Towards Efficient Audio-Visual Source Separation and Synthesis

Juan Felipe Montesinos

TESI DOCTORAL UPF / year 2023

THESIS SUPERVISOR
Gloria Haro
Department Department of Information and Communications Technologies

Insights and acknowledgments

I would like to start talking about “the PhD”. Before starting it, I though doing a PhD was about being a smart guy with clever ideas. Then, I realized it has many more dimensions and factors. The success is not only about you, but maybe, about having good reviewers, or computational resources. Maybe it is about having the right people around, or having an idea in the right moment. Funny fact, deep learning (backpropagation to be technical), the technology called to change the present and the future, was invented in the 60s. The success could be a matter of appearing in the queue of Arxiv, or being cited in a tweet or a paper. In short, the PhD, for better or worse, is a whole environment with many dimensions interlaced.

I really believe, despite I’m very rational, that life guides us. I met my PhD supervisor, Gloria Haro, back in 2017, while carrying out my MSc. in computer vision. She was teaching 3D vision and she was considered as great teacher among us, the students. By the end of my masters, I was supposed to travel to Paris to carry out my master thesis in a well-positioned company. The day before flying, I was told the university and the company couldn’t reach and agreement, my final project was canceled. That was a tough moment, I was disoriented but, as I mentioned, life guides us. Gloria offered me to carry out the master thesis with her, which led to this PhD afterward. From my point of view, Gloria has been a wonderful supervisor, she is very supportive and understanding, and works tirelessly. Truly a role model. My master thesis was supervised as well by a PhD student, Olga Slizovskaia, who later became my colleague. Same as Venkatesh Kadandale, another good colleague whom I met by the beginning of the PhD. Despite the insightful research talks we may have had, I really appreciate having someone who understands the very specific frustration that audio-visual field and ML entails. Sometimes we forget mental health is a thing. Likewise, I cannot forget to mention Patricia Vitoria and Adrià Arbués who always were up for a coffee or bringing funny moments that made the lab to be alive. I’m also grateful to have had the opportunity to collaborate with Daniel Michelsanti at Oticon, in Denmark. Everyone at Oticon were very friendly, which made it a great

experience.

This thesis was very conditioned by the COVID pandemic. Virtual conferences couldn't recreate the human perspective of on-site conferences: discovering new cultures and countries, networking, meeting new different people that may give you support, insights, opinions or new points of view about topics one would never wonder. However, facing such hard times helped me to learn about myself and probably changed my life, my mindset, and my way of seeing the world.

Lastly, taking a look back at my childhood, I feel I need to dedicate a small honorific mention to my mother, Pilar Garcia. She raised me and supported me, following the idea that education is the best gift a person can be given.

That being said, I find the PhD has been a great experience. No matter how successful or not it can be considered by society, it is now an important part of who I am, and I wouldn't change that.

Abstract

Our brain has the innate capability of isolating different sounds in noisy environments (the cocktail party problem), as well as understanding the relationship between what we see and what we hear. This thesis aims to bring these human cognitive skills to computers by contributing to the improvement of speech, singing voice and music sound source separation as well as speech inpainting. To do so, we explore new video representations and their suitability for the aforementioned tasks. In case of audio-visual voice separation, we used face landmarks, which encode motion and drop appearance. This allows developing lightweight, real-time audiovisual sound source separation systems. We show how visual information can be beneficial in noisy or multivoice environments, and we propose a deep neural network competitive to state-of-the-art models that exploit both, motion and appearance.

Speech inpainting can be seen as an extreme case, when the accompaniment sources are so noisy that no signal can be recovered or the speech signal is corrupted. We show how deep-learning-based visual embeddings extracted from large-scale models encode enough information to reconstruct long gaps of speech, up to 1.6s. We also show how the audio-visual models do outperform audio-only systems.

Resumen

Nuestro cerebro tiene la habilidad innata de aislar diferentes sonidos en ambientes ruidosos (Efecto de fiesta de cóctel), así como de entender la relación entre aquello que vemos y oímos. Esta tesis tiene como objetivo trasladar estas habilidades cognitivas, características de los humanos, a los ordenadores. De esta forma, se busca contribuir a la mejora de, por un lado, la separación de sonidos tanto en el ámbito de los discursos hablados, como en el de la música y la voz cantada; y por otro a la reconstrucción contextual de discursos hablados. En el caso de la separación de discursos hablados, usamos marcadores faciales que codifican el movimiento y dejando de lado la apariencia. Esto permite desarrollar sistemas audiovisuales de separación de sonidos en tiempo real y ligeros. Asimismo, mostramos como la informacion visual puede ser beneficiosa en entornos ruidosos o con múltiples voces, y proponemos una red neural competitiva con modelos del estado del arte que utilizan movimiento y apariencia.

La reconstrucción del discurso hablado puede ser visto como un caso extremo, cuando las fuentas que acompañan al discurso son tan ruidosas que este no se puede recuperar, o cuando la propia señal está corrupta. En este escenario mostramos como representaciones de la información visual extraídas con modelos de gran tamaño codifican suficiente información para reconstruir largos segmentos de discurso hablado, de hasta 1.6s. También mostramos como los modelos audiovisuales superan a los modelos que sólo utilizan audio.

Contents

List of Figures

List of Tables

List of abbreviations

AO Audio Only. 98

AV Audio-Visual. 2

AVSI Audio-Visual Speech Inpainting. 7

AVSS Audio-Visual Source Separation. 7

BERT Bidirectional Encoder Representations from Transformers. 28

BSS Blind Source Separation. 28

CASA Computational Auditory Scene Analysis. 33

CNN Convolutional Neural Network. 8

DFT Discrete Fourier Transform. 19

ELU Exponential Linear Unit. 103

FiLM Feature-wise Linear Modulation. 8

GCN Graph Convolutional Network. 27

GELU Gaussian Error Linear Unit. 79

ICA Independent Component Analysis. 33

LR Learning Rate. 47

MAE Mean Absolute Error. 103

MHU-Net Multi-head U-Net. 45

MIR Music Information Retrieval. 38

ML Machine Learning. 4

MLP Multi-Layer Perceptron. 100

NLP Natural Language Processing. 6

NMF Non-negative Matrix Factorization. 33

NN Neural Network. 6, 7

PLCA Probabilistic Latent Component Analysis. 33

RNN Recurrent Neural Network. 78

ROI Region of Interes. 10

SAR Sources to Artifacts Ratio. 49

SDD Solid State Drive. 9

SDR Source to Distortion Ratio. 49

SIR Source to Interferences Ratio. 49

SoP Sound of Pixels. 45

SOTA State of the Art. 14

ST-GCN Spatio-Temporal Graph Convolutional Network. 14

STFT Short-Time Fourier Transform. 19

Chapter 1

INTRODUCTION

1.1 Motivation

Once Rene Descartes wondered if the world was real or just an illusion created by an evil genius. Likewise, we would like to reflect on how do we perceive the world.

Most events that take place in our world generate a rich source of visual and auditory signals. Imagine having dinner in a cozy table with friends and family. All of a sudden, a bottle falls onto the floor in the other side of the table. If we just use the sight, we cannot know whether the bottle has broken or not, since the table occludes the event, or even what the bottle is made of. If we just use the hearing, we can perceive whether a piece of glass or plastic has broken or fallen, but we would not be able to identify which object was it.

Acoustic and visual signals are two counterparts from the same event in the real word. That is why, our perception is highly multimodal. Audio-visual (AV) signals can help to disentangle ambiguities: distinguishing between a dog or a wolf, whether a cup is made of glass or plastic or to track and localize better [Charbonneau et al., 2013, Hofman et al., 1998]. There are even more constrained cases, e.g., to infer the distance of a thunder, we need both the visuals and the acoustics of the event. These examples aim to illustrate how acoustic and visual signals are often complementary and can improve our perception. It is not surprising that the human cognitive system evolved to learn efficiently from multisensory stimuli [Shams and Seitz, 2008].

Among all natural phenomena, speech is one of the most relevant ones for humans. We are constantly exposed to speech on the TV, on social networks, at work, at home... Speech is so important that, through evolution, humans developed special areas in the brain devoted to the production and comprehension of natural language [Dronkers and Ogar, 2004]. Unsurprisingly, speech is not processed as an acoustic signal by the brain, but as an audio-visual one. In [Chen, 2001] the authors show how certain phonemes (the smallest phonetic unit in a language that is capable of conveying a distinction in meaning) that are hard to distinguish acoustically can be distinguished audiovisually. In [Fisher, 1968] the authors illustrate

how, for a given phoneme, the perceived sound can change depending on the visuals from the lips.

We can trace the origins of speech back to singing voice, which is related to the origin of the voice. The most accepted theory is speech appeared as an evolution from singing voice, understood as the "vocalization" to express emotion and imitate natural sounds, as explained in [Aiello and Dunbar, 1993, Jespersen, 2013]. Even though both, singing voice and speech share some characteristics, e.g., intelligibility, they are substantially different. Whereas speech aims to transmit information and certain emotions (anger, sadness, happiness...), singing voice also includes melody and rhythm, pursuing aesthetical sounds in addition to content. For this reason, singing voice is often ornamented with resources such us vibrato or humming, originating a great amount of sustained vowels and notes.

Singing voice is intimately tied to music. Music shares its origin with singing voice, and both have played an important role in our evolution [Schulkin and Raglan, 2014]. That may be the reason voice is considered as the original instrument. At first look, we may think of music as unimodal event, however, there are studies showing that the nature of music perception is multimodal [Leman, 2017, Thompson et al., 2005]. Music is an AV performance, and it is intrinsically related to dance. Worldwide, folkloric music and tribal songs are usually paired to different dances. Even in our days, all the modern songs are usually released as video clips, not just audio clips. And many live performances involve visuals: instrumental concerts, ballet, music concerts, films... In fact, we prefer AV performances over audio-only ones. [Platz and Kopiez, 2012] concludes we score higher AV performances in terms of overall impression, expressiveness, likability or quality.

Due to the aforementioned reasons, we have spent decades developing systems which can capture these audio-visual streams as we perceive them, with the highest fidelity. With the democratization of these devices and the expansion of internet, billions of hours of recordings and generated content are available in public platforms such as YouTube, Twitch, Vimeo, Instragram... Unfortunately, proper systems to exploit all this in-

formation were not developed in parallel due to lack of computational power and effective techniques. More recently, with the rise of machine learning (ML), researchers and companies started developing effective and automatized tools to make profit from all this data. Initially, ML was focused on supervised learning. Supervised learning algorithms assume access to human-labeled datasets to train ML models. This scenario was feasible for computer vision algorithms involving images, as humans can understand images without any training; but it is prohibitively expensive for video, where the amount of information to be labeled is huge. It is also unfeasible for audio, where, unlike images, humans are cannot interpret their representations (waveforms, spectrograms etcetera...) without training and expertise.

In order to process these millions of hours of video, other kinds of algorithms that do not require fully supervision, i.e., human-labeled data, must be developed. Inspired by how humans can learn autonomously, this thesis aims to find different ways of learning from audio-visual content and the natural correlation between audio and video to train deep learning algorithms with few or no human-labeled data. More specifically, this thesis contributes with new self-supervised audio-visual deep-learning-based models for speech inpainting and for source separation in different contexts: musical instruments, singing voice and speech.

AV speech comprehension, separation, and enhancement play an important role. AV Voice separation can help in speaker diarization and transcription. For example, in talk shows where several people speak at the same time. Or in parliamentary sessions, it would be possible to track each speaker's speech, obtaining clean isolated recordings and transcribing them. The same applies to a news broadcast from the street, or for multilanguage doubling of speeches, where the journalist's voice could be enhanced for viewer's convenience. In conferencing systems, background noises or leaking voices which do not correspond to the speaker can be removed. In the worst case, for applications where audio is totally unusable, speech reconstruction from lips can substitute this enhancement/separation process.

Regarding music and singing voice, isolating different instruments or

voices can be beneficial for transcription and educational purposes. It could also be applied in the context of remixing, where each source can be isolated, and the composition can be resynthesized with different filtering, tempo, or sources.

1.2 Audio-Visual in Machine Learning: Introduction and Challenges

1.2.1 Introduction to Multimodal Analysis

As mentioned in Sec. 1.1, AV music and voice separation, as well as AV speech inpainting can be tackled with self-supervised algorithms.

The goal of self-supervised techniques is to train ML algorithms without any human-annotated data. In unimodal domains, e.g., vision, natural language or audio, a typical procedure is reconstructing, partially or totally, the input data. Autoencoders are a good example of self-supervision [Bank et al., 2020, Cinelli et al., 2021], where a neural network (NN) maps the input data to a low-dimensional space (compression) for then reconstructing it (decompression). Or inpainting, which is a traditional self-supervised task in vision, where an image is masked partially to infill the masked content coherently with the rest of the image. Transformers [Vaswani et al., 2017] are one of the most powerful architectures to process sequences. They can make profit from *mask-and-reconstruct* schedules in several fields such us Natural Language Processing (NLP), audio, or AV [Devlin et al., 2018, Hsu et al., 2021], improving performance, generalization, and robustness. This idea will be detailed in Sec. 2.2.2.

In a multimodal scenario, possibilities beyond self-reconstruction appear. The key idea is to exploit the relationship among modalities as a supervision cue. In the AV field, these are the correspondences between audio and video in recordings. A good example is cross-modal estimation, where the goal is to infer audio from video or the other way around, e.g., lip-to-speech estimation [Mira et al., 2022], or the aforementioned *mask-and-reconstruct* pre-training technique [Shi et al., 2022]. Similarly, cross-modal correspondence aims to solve whether an audio and video streams match or not [Arandjelović and Zisserman, 2018]. If audio and video streams are finely correlated, AV synchronization can be estimated [Kadandale et al., 2022, Truong et al., 2021, Chen et al., 2021]. Other interesting tasks which involve correspondence are cross-modal style trans-

fer [Li et al., 2022] or zero-shot learning [Mercea et al., 2022], where audio, video, and text are mapped into a common latent space.

Lastly, it is worth mentioning some important multimodal tasks, though these are fully supervised: text-to-speech [Hsu et al., 2021], or AV speech recognition [Shi et al., 2022].

In the context of this thesis, there are works addressing AV music source separation, such as [Zhu and Rahtu, 2020], or speech separation [Gao and Grauman, 2021] among others. However, AV singing voice separation was not explored in the literature before. AV speech inpainting (which is similar to lip-reading) was already explored for small datasets in [Morrone et al., 2021]. A detailed literature review of both, AV source separation (AVSS) and AV speech inpainting (AVSI) is carried out in Secs. 4.2 and 5.2; and Sec. 6.1 respectively.

Despite the rich variety of tasks to solve, there are certain challenges that motivated and constrained the scope of this thesis. In the next section, we will describe them.

1.2.2 Challenges

The curse of multimodal

A really common problem in multimodal tasks is that one modality tends to dominate over the other [Michelsanti et al., 2021, Wang et al., 2020]. For example, in the AV speech recognition task, where the goal is to transcribe AV speech recordings, audio is often enough to solve the task, and it is the most related modality to text. Learning to extract meaningful information from the visuals is slower than learning from audio, thus, visuals are ignored. This has led to different ways to fuse AV information, often inspired by biology; and strategies to force NN models to attend both modalities. Regarding the fusion stage, we can classify existing strategies into:

- **Early fusion,** where the acoustic and visual modalities are concatenated as raw or roughly raw data (e.g. [Morrone et al., 2021]).

- **Mid fusion,** where the acoustic and visual modalities are first processed independently for later on being fused and processed jointly (e.g. [Owens and Efros, 2018]).

- **Late fusion,** where the acoustic and visual features are processed separately and the information is shared via metric learning (e.g. [Korbar et al., 2018])

Regarding the fusion mechanism, there are three main groups: by using attention [Chen et al., 2021], by concatenation [Morrone et al., 2021] or using Feature-wise Linear Modulation (FiLM) [Perez et al., 2018].

Nevertheless, the most effective way depends on training strategies rather than fusion mechanisms. In [Shi et al., 2022], the authors use modality dropout and a *mask-and-reconstruct* procedure, forcing the network to use the visual modality to accomplish the task. Other strategies can be adding noise to the audio, and the most effective one, posing the problem in such a way that visuals are required to solve the task, as in [Gabbay et al., 2018]. In our case, when working in AVSS, we mix two human voices within the same audio track such that the network must pay attention to motion to figure out the target voice, as we will show in Chapters 4 and 5.

Computational cost

Neural networks suffer to process high-resolution content. 4K (4090 x 2160) and full HD (1920 x 1080) are the standards in video resolution as of today, while NNs usually work with resolutions as small as 64×64, 112×112 or 256×256. Despite some works are starting to close the gap, this is still a big issue in the computer vision community. Calculating coarse numbers for a 25-fps RGB video implies approximately 120 Mbps to process. In case of multimodal information, we require extra 512 kbps corresponding to a 16 kHz audio signal.

There are two common types of NN widespread in image processing. 2D convolutional neural networks (CNNs) and transformers. Their extension to video is trivial. In the former case, convolutional kernels

are extended, leading to 3D CNNs. These architectures are data greedy, as there is a big increment in the amount of parameters from 2D CNNs to their 3D counterparts (see Table 1.1), although hybrid solutions appeared [Leong et al., 2020, Tran et al., 2018]. In the latter case, no spe-

Model	2D-CNN	3D-CNN
VGG-16	134.7 M	179.1 M
ResNet-18	11.4 M	33.3 M
ResNet-34	21.5 M	63.6 M
ResNet-50	23.9 M	46.4 M
ResNet-101	42.8 M	85.5 M
ResNet-152	58.5 M	117.6 M
DenseNet-121	7.2 M	11.4 M
DenseNet-169	12.8 M	18.8 M

Table 1.1: Number of training parameters in millions (M) for VGG, ResNet and DenseNet models [Leong et al., 2020].

cial adaptations are required. Simply, video can be considered as a longer sequence. This is problematic as transformer's computational complexity is $\mathcal{O}(n^2)$ where n is the sequence length. In any case, the computational cost of both, CNNs and transformers, grows exponentially with the spatial and temporal resolution.

Storage Cost

Besides computational complexity, there is an additional cost that is not usually taken into account. Storage cost. Deep learning requires really optimized pipelines, which often forces using high-end hardware and uncompressed data formats. Working with video often requires long preprocessing pipelines to obtain curated data. A typical pipeline is described in Pipeline 1.

In the hard disk context, Solid State Drive (SDD) disks are a must [Kadve, 2016]. Nowadays, SSD disks' capacity varies from 128 Gb to 8 Tb, being 1 Tb the most extended size. Giving a glimpse about the

Pipeline 1 AV-Dataset Pre-processing

1: Downloading the audio-visual recordings.
2: Video re-encoding (to correct missing frames and have a homogenous frame rate).
3: Audio resampling.
4: Frame-wise detection of the region/s of interest (ROIs).
5: Trimming
6: Cropping around the ROIs.
7: Image alignment (warping).
8: Video resizing.
9: Video compression (if not using raw information)

required storage capacity, one of the standard datasets in the AV speech community, *Voxceleb2* [Chung et al., 2018], requires more than 1 Tb **once curated**. Moreover, *Voxceleb2* is a +2000-hours low-quality dataset conformed by YouTube videos. High-quality AV datasets, such as MODALITY dataset [Kawaler and Czyżewski, 2019] ($\approx$ 30h), requires around 213 Gb of hard disk before being curated. In conclusion, working and pre-processing AV datasets is often intractable for many research groups and universities, as this would require specialized knowledge (distributed storage and computing...) and lots of storage resources.

Data Curation, Data shortage and Data Quality

A key ingredient to bound computational and storage cost is data curation. For example, on the TV news, we usually see a presenter in a television studio. If we are analyzing the speech content, e.g., to carry out AV speech recognition, we do not care about the studio but the presenter and, often, just his/her face or mouth. Therefore, the mouth can be considered as the ROI. As shown in Pipeline 1, step 4, we usually find per-frame ROIs, then crop and trim the video to finally align all the frames. This is crucial to reduce the overfitting and both, the storage cost and computational cost (cropped videos are much smaller).

However, this step is not straight-forward in many cases. For human-

related content, there are dozens of algorithms to track the body pose, the hands, or the face, even to extract depth maps or 3D reconstructions. For everything else, the best we can usually find are object detectors. Still, the classes are limited. As a consequence, there is a strong data shortage in many different scenarios. Without robust algorithms, it is not possible to create automatized large-scale datasets from video platforms. Even if it is still possible to gather data manually, in small-scale video datasets (less than 50 hours), a lack of ROIs may lead to overfitting.

Another relevant problem is data quality. Many popular large-scale video datasets have poor quality. This is problematic, as noisy samples are related to the double descent phenomena [Nakkiran et al., 2020]. In small-scale datasets, this noise may prevent NNs from learning useful patterns.

The data issues mentioned in this section affect specially the AV community focused on music analysis. While there are some efforts to increase the available data, there is still no suitable large-scale AV musical dataset. To this extent, this thesis contributed to the community with the creation of two different datasets: *Acappella* dataset, an AV dataset of people singing *a cappella* [Montesinos et al., 2021]; and *Solos* , a dataset of musicians solo playing different musical instruments [Montesinos et al., 2020].

Reproducibility

Reproducibility in ML and, concretely, in the AV field, deserves a few words. Despite computers are deterministic, there are several reasons why this is challenging in ML. First, datasets are huge and difficult to track. In an ideal world, we should keep a copy of each dataset once we finish an experiment and process the results. The reality is datasets mutate through time, hence this is unfeasible. That is the case in the AV community, where many datasets are a compilation of YouTube IDs. This provokes further researchers not to have access to exactly the same data, as certain recordings may have been removed. At the time of writing this thesis, one of the most important speech datasets, *Voxceleb2* [Chung et al., 2018],

which was available through direct download, was replaced just by the corresponding YouTube links due to copyright issues.

Another remarkable problem is the traceability of the samples. As there are no official training/testing splits, researchers tend to make their own. This process often involves splitting recordings into chunks which are not clearly traced.

Ethics and data privacy

Due to the delicacy of the subject, we will omit any reference in this section, as the purpose is to reflect certain issues rather than pointing towards specific cases.

Large scale ML models need huge amounts of data. To respond to the growing demand for data, many services, and companies collect user data under abusive terms and conditions with fine print. That is the case for many face swapping services, voice conversion services, smart vehicles equipped with cameras and sensors, coding assistants and a great amount of phone apps including social networks or cloud storage services. A great culprit is the fact many services are cloud based, forcing users to send their personal data, a moment in which they lose any control or track on it.

Another relevant problematic is ML models may exploit data biases and spurious correlations. For example, we can imagine models predicting the probability that a person commits a crime based on his appearance. Or models making use of correlations based on ethnicity in topics such us medical diagnosis, AV tasks involving face or voice synthesis, etcetera...

Lastly, it is worth mentioning generative models are reported to generate training samples and samples with copyright. It is even possible to recover training samples or private information targeting specific samples from trained NNs, as shown in different works in the literature, such as [Haim et al., 2022, Song et al., 2017], or [Wang and Kurz, 2022].

In this thesis, face landmarks and motion embeddings are used in replacement of raw video, which alleviates the data privacy by dropping the appearance and user identity. This idea will be detailed in Sec. 2.3.1.

1.3 Thesis Scope and Contributions

Once pointed out the problematics of ML and the AV field, let us define the scope of the doctoral thesis. As mentioned in the introduction, this thesis focuses on the development of self-supervised algorithms for processing speech, music and singing voice AV signals. Concerning the above challenges, the goals of the thesis are the following:

1. The design of different NN architectures regarding four pillars: latency, performance, self-supervision and real-world applicability. These architectures involve the following tasks:
   (a) Music Source Separation
   (b) Singing Voice Separation
   (c) Speech Separation
   (d) Audio-Visual speech inpainting
2. The creation of datasets that benefits the AV community and alleviates the data shortage, as well as optimal data-loading pipelines to train NNs efficiently.
3. Developing software, libraries, and frameworks which provide tools to work with AV data in a reproducible manner.

This thesis contributed to the development of the audio-visual source separation and audio-visual speech inpainting problems in the following ways, which are sorted historically:

1. **AV music source separation**: We propose a collection, development, and preprocessing of an audio-visual dataset of musicians playing solo excerpts based on YouTube videos. We provide Youtube IDs and the corresponding timestamps of relevant segments of each video, body and hand skeletons extracted with an open-source library, Openpose [Cao et al., 2019]. This work was developed in [Montesinos et al., 2020].

2. **AV singing voice separation**: We address the task of singing voice separation. Due to the lack of public datasets, we collected a 46-hour dataset of *a cappella* solo singing videos in four languages. We designed a source separation model that is based on a U-Net architecture conditioned on motion features. Motion features are extracted with a Spatio-Temporal Graph Convolutional Network (ST-GCN) that receives a sequence of face landmarks as input. We show the superior performance of face landmarks compared to video in small datasets, where overfitting is problematic. Besides, we study the relevance of language in AVSS, showing that NNs can perform similarly for unseen languages. Lastly, we show how audio-visual models for singing voice separation are particularly useful when there are multiple voices in the mixture, or when the volume of the target voice is low compared to the other sources. This work was carried out in [Montesinos et al., 2021].

3. **AV speech separation**: We develope a new audio-visual architecture based on transformers and ST-GCNs that is State of the Art (SOTA) in speech and singing voice separation, showing how landmarks can be competitive against video-based networks on large scale datasets without exploiting appearance attributes. Furthermore, we study different transformer variants to efficiently consume audio-visual data. Besides, we propose a two-stage training solving two different optimization problems, which boosts the results in terms of interferences from other sources at a cheap cost. Lastly, we study the suitability of models trained in speech datasets for singing voice, showing a drastic drop in performance, which indicates the necessity of gathering more singing voice data. This exploration was done in [Montesinos et al., 2022a].

4. **AV speech inpainting**: We propose a state-of-the-art transformer-based architecture that is capable of reconstructing long gaps of corrupted audio by leveraging the visual information. Visual features are extracted with AV-HuBERT [Shi et al., 2022], which encode information at a viseme level. We study the degradation of the

generated speech for both AV models and audio-only models. To this extent, we enlarge the gap duration, showing audio-only models collapse while AV models quality is consistent. This work was carried out in [Montesinos et al., 2022b].

1.4 Outline of the thesis

Chapter 1 introduces the reader to the world of multimodal processing in machine learning, briefly revisiting the multimodal perception from a human perspective and its impact on society and evolution as motivation to the problems tackled in the thesis: AV music and voice separation as well as AV speech inpainting. We also overview the challenges in the field, which have constrained the thesis and bounded its scope. The chapter is closed by the thesis' academic contributions by topics. In Chapter 2, we provide the mathematical and theoretical tooling to understand the acoustic and visual data representations, the transformations used in this thesis, and foundations on source separation and graphs. Moreover, we explain the main NN's architectures used, ST-GCNs and transformers.

Afterward, we present two different blocks that cover the main topics of the thesis: the first block, devoted to the AVSS problem and includes Chapters 3-5. Chapter 3 presents *Solos*, the dataset of AV music performances. Chapter 4 presents *Acappella* dataset and explores singing voice separation using face landmarks and graphs. Lastly, Chapter 5 addresses AV voice separation using transformers. The second block consists of a single chapter, Chapter 6, which devoted to multimodal speech reconstruction. The thesis ends with a chapter summarizing its content, the conclusions, and proposing future work.

Chapter 2

THEORETICAL BACKGROUND

2.1 Audio representations

2.1.1 Waveforms

Sound waves are longitudinal mechanical waves that travel through a physical medium, usually air. As longitudinal waves, there exist compression and rarefaction stages, where the separation among air particles is larger or smaller than in average. Measuring the pressure allows characterizing waves, as shown in Fig. 2.1. These signals were standardized, resulting in different waveform formats [IBM, 1991]. Nowadays, waveforms are the *de facto* audio representation in signal processing.

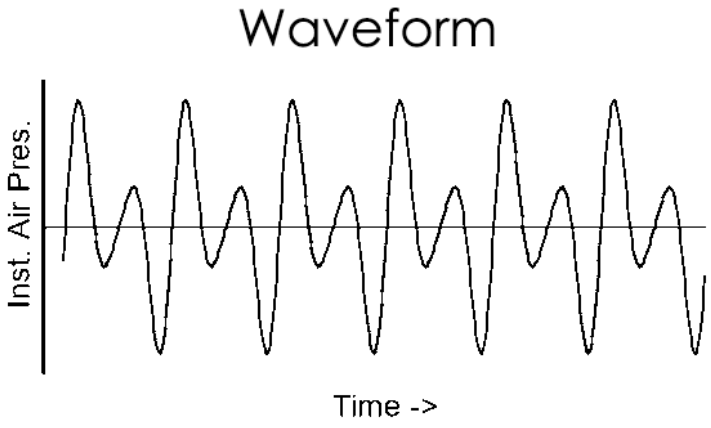

Figure 2.1: Time representations of a periodic sound wave. Illustration from [Kiper, 2016].

2.1.2 Time-frequency representations: Short-time Fourier Transform

Waveforms are a very low-level representation of sound waves, hardly interpretable by humans. A more intuitive and compact representation of

acoustic information is the time-frequency characterization of waveforms. In time-frequency characterization, the signal is split into short segments to carry out a frequency analysis on each of them. These representations assume signal properties vary slowly in time, i.e., they are approximately constant within each segment. For example, in case of speech, a phoneme duration is 80 ms in average. Some of the most relevant representations are Constant-Q transform, Wavelet transform, Wigner Distribution Function, Discrete Cosine transform, or Short-time Fourier Transform (STFT). STFT is specially suitable for source separation, as it is a linear invertible transformation based on the well-known Fourier Transform. That is why, it is widely used in the ML literature and will be the main audio representation in this thesis.

The STFT of a 1D signal can be calculated as a sequence of Fourier Transforms carried out on a windowed signal [Diniz et al., 2010]. Let $x(t)$ be a 1D continuous signal and $X(\omega, \tau)$ be the corresponding STFT. Then, STFT can be computed as:

$$X(\omega, \tau) = \int_{-\infty}^{\infty} x(t) g(t - \tau) e^{-j\omega t} dt$$

where $g(\tau)$is the window function.

In practical applications, discrete-time STFT is used. We refer to [Benesty et al., 2008] for detailed information. Assuming now $x(t)$ is finite, we can define its discrete version, $x[t]$, by sampling the signal at a frequency F_s. Data segments (called frames) are extracted by sliding a finite window, $g[n]$, at regular intervals. Each frame, x_l can be defined as $x_l[n] = g[n]x[n + lL]$, where l is the frame index, n is a local time index (relative to the window), L is the hop length and N is the window length. Then, the discrete-time STFT, $X[k, l]$ can be constructed by stacking the Discrete Fourier Transform (DFT) of each frame x_l. Yielding to the expression:

$$X[k, l] = \sum_{n=0}^{N-1} g[n]x[n + lL]e^{-j2\pi nk/K}$$

where k and l indicate, respectively, a frequency index, and K denotes the DFT size.

The magnitude component of STFT, $\|X[k,l]\|$, or its spectrogram, $\|X[k,l]\|^2$ are easily interpretable. In Fig. 2.2, it can be seen the spectrogram of a speech waveform. The horizontal patterns correspond to harmonic frequencies of the voice, which is characteristic of each person.

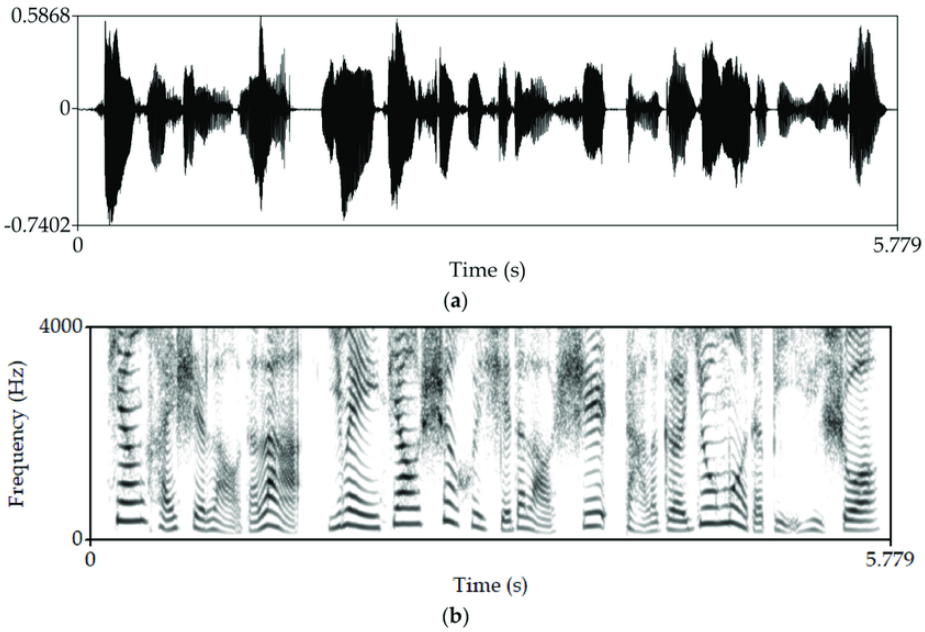

Figure 2.2: a) Speech waveform and b) its corresponding spectrogram. Illustration from [Lu et al., 2018].

2.1.3 The Human Auditory System and Mel Spectrogram

Sounds we can hear depend on their intensity and spectrum, and our auditory system. STFT, while suitable for ML applications of any kind, is a pure mathematical analysis of the spectrum of a waveform, and does not take into account the human auditory system and mechanisms behind

sound perception. Our auditory system is highly non-linear. Different frequencies require different levels of energy to be perceived. Besides, frequency perception is interlaced, as described in [O'shaughnessy, 1987]: "perception of sound energy at one frequency is dependent on the distribution of sound energy at other frequencies as well as on the time course of energy before and after the sound".

Mel scale is an analytical expression of this non-linearity, developed by measuring listeners' perception to form an equidistant distribution of pitches. Mel scale is depicted in Fig. 2.3. We can observe that humans perceive low frequencies with a higher resolution.

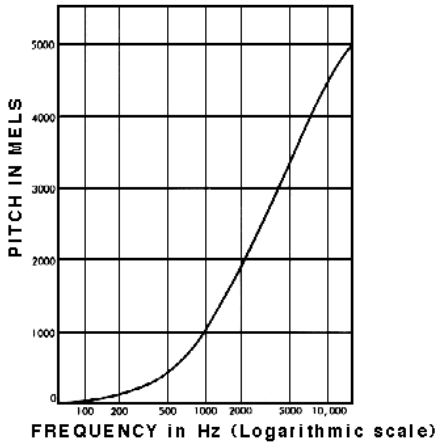

Figure 2.3: Mel scale. Illustration from [Appleton et al., 1975]. The Mel scale's analytical expression is $2595 \log_{10}(1 + f/700)$, where f is frequency (Hz).

Built upon STFT and the aforementioned psychoacoustic findings, mel spectrograms were developed. Mel spectrograms are nothing but a dimensionality reduction carried out with a mel-frequency filterbank. The relationship between a mel-spectrogram, Y, and its former spectrogram is defined by a matrix multiplication, $Y_{pl} = \sum_k W_{pk} \|X_{kl}\|$, where W is the mel filterbank.

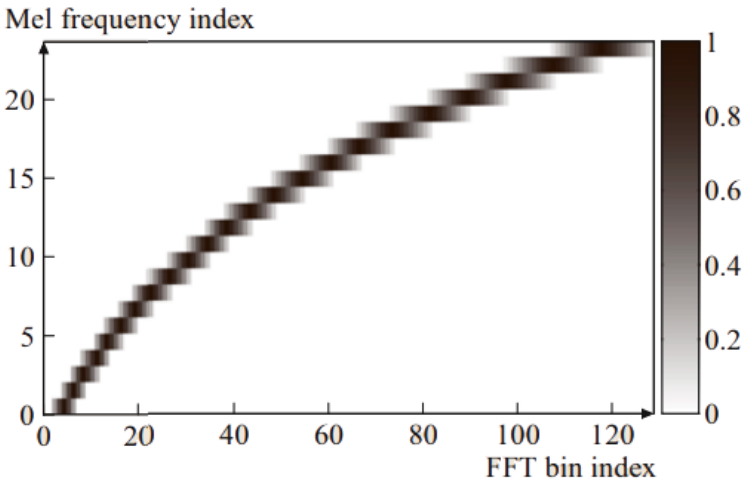

Figure 2.4: Mel filterbank (23 filters) as a function of STFT with DFT size 128. Illustration from [Benesty et al., 2008].

2.2 Deep neural networks

2.2.1 The U-Net

The U-Net architecture is a type of CNN that is commonly used for tasks such as image segmentation, image-to-image conversion, audio source separation and others. The U-Net architecture was first introduced in [Ronneberger et al., 2015] for medical image segmentation and owes his name to its characteristic U-shape, as illustrated in Fig. 2.5.

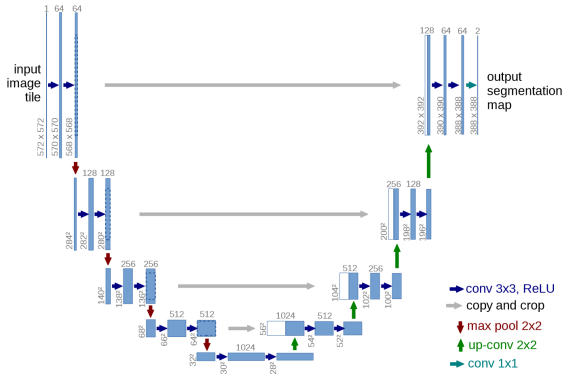

Figure 2.5: Original U-Net architecture [Ronneberger et al., 2015].

The U-Net consists of a contracting path and an expanding path, that act as encoder and decoder, respectively. The contracting path follows the typical architecture of a CNN, with alternating convolutional and pooling layers that progressively reduce the spatial resolution of the input. In the expanding path, the spatial resolution of the feature maps is increased through the use of up-sampling layers. The up-sampling layers are concatenated with the feature maps from the corresponding layer in the contracting path via skip connections, allowing the network to use both high-level semantic information from the contracting path and low-level spatial information from the input image. Different modifications have been proposed in different fields and context, for example using strided convo-

lutions instead of max-pooling.

2.2.2 Transformers

A transformer is a type of NN designed as a sequence-to-sequence model. It maps sequences of any kind and length into other sequences of any kind and length. That is why it has been used in translation, text-to-speech, or image classification among other tasks.

The core idea behind transformers is self-attention and multi-head attention. Self-attention is a mechanism used to calculate the relevance of each element in the input sequence with respect to all other elements in the sequence. This is done by projecting the input sequence onto three different vectors called keys, queries, and values; and calculating the similarity between each query vector and all key vectors. The resulting similarity scores are used to compute a weighted sum of the value vectors. Multi-head attention is an extension of self-attention in which multiple self-attention mechanisms are applied to the input sequence in parallel, and the outputs of these mechanisms are concatenated and projected onto a final output vector. This allows the model to learn multiple different representations of the input sequence simultaneously.

The mapping capabilities of the model are given by the "feed-forward" layer, which is nothing but a multi-layer perceptron applied after attention modules.

There are several advantages to using transformer models in machine learning. Some key advantages of transformers include:

- Improved parallelization: Since the self-attention mechanism calculates the relevance of each element in the input sequence with respect to all other elements in the sequence at once, it can be parallelized. This allows transformer models to be trained and evaluated much faster than recurrent neural networks.

- Better performance on long sequences: Transformer models are particularly well-suited to processing long sequences of data. This is because self-attention can capture long-range dependencies within

the data from the very beginning. For example, in image classification, CNNs need depth to reach a proper receptive field; and in text classification, recurrent NNs tend to forget previous elements of sequence through time.

- Better handling of variable-length inputs: The length of the input sequence may vary from one sample to the next. As transformer models do not rely on fixed-sized input representations, they can easily handle variable-length inputs without the need for padding or truncation.

Overall, the use of transformer models in ML can provide significant performance benefits, particularly when working with long sequences of data or when dealing with variable-length inputs.

2.3 Visual representations

2.3.1 Motion, graphs, and face landmarks

Face landmarks are a set of keypoints in a facial image that correspond to important anatomical features, such as the eyes, nose, mouth, and jawline as shown in Fig. 2.6. Face landmarks can be considered as a graph consisting of a set of edges, E, and nodes, V, such that $G = (E, V)$. V can be defined as $V = \{v_{it} | i = 1..N; t = 1..T\}$ where i represents the spatial index at the t-th video frame. The set of edges can be decomposed into temporal edges and spatial edges. Spatial edges are defined as $E_s = \{v_{it}v_{jt} | (i, j) \in \mathcal{H}\}$, where $\mathcal{H}$ is the set of indices that retains the face shape. Temporal edges connect each landmark with the analogous one in the next frame as follows, $E_f = \{v_{it}v_{i(t+1)}\}$. This representation is suitable for processing face landmarks with spatio-temporal graph neural networks, which are very good at extracting motion information.

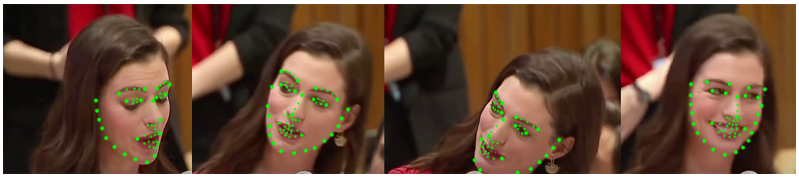

Figure 2.6: Example of landmarks extracted from a high-quality video with large head displacements and broad points of view at 1 FPS. Frames correspond to Anne Hathaway's speech: *Paid Family Leave*.

There are many associated advantages in using face landmarks, which alleviate partially or totally many of the challenges mentioned in Sec. 1.2.2. In Table 2.1, the storage cost of different common video standards is shown. The storage cost grows quadratically with video resolution, and so does the computational cost to process each video. Due to the huge amount of information video encodes, we may experience overfitting problems, storage shortage, or lack of computational power. The key to solving these is landmarks are a compact and efficient representation

Video Standard	Resolution	Typical resolution of a speaking face	Amount of uint8 values	Storage Cost (kB)	Landmarks resolution	Storage Cost (kB)
HD	1280×720	256×256	197k	192	68×3	0.2
Full HD	1920×1080	512×512	786k	768	68×3	0.2
QHD	2560×1440	700×700	1.5M	1436	68×3	0.2
4K	4096×2160	1024×1024	3.1M	3072	68×3	0.2
8K	8192×4320	2048×2048	12.6M	12288	68×3	0.2

Table 2.1: Most common standards in video for 17:9 and 16:9 formats and the storage cost of a single frame. The typical resolution of speaking face is based on a single speaker giving a speech from a tribune. We refer to Anne Hathaway's speech: *Paid Family Leave* as an example. The video can be found at `https://www.youtube.com/watch?v=gkr57P0fwbI`

of a person's face. First, this can make it easier to train NNs and can reduce the amount of data that needs to be processed, which can speed up the training process and reduce overfitting. Second, as we can represent face landmarks as a graph, we can use Graph Convolutional Networks (GCNs). Unlike images, where the distance among pixels is fixed and structured, the distance among the coordinates of face landmarks can vary. Hence, GCNs can effectively capture the spatial relationships between different nodes with few data. Third, face landmarks drop appearance and background information. Therefore, they are robust to variations in lighting and pose, and ensures only pure motion-based features are learned. This increases the models' privacy, as no additional image information is required. And generalization in terms of gender and ethnicity. Lastly, the aforementioned face alignment also reduces overfitting and increases generalization to unknown camera's points of view, which is particularly useful if working with few data.

2.3.2 Audio-visual HuBERT

While face landmarks are a well-suited representation for audio-visual speech tasks, they are still low-level features, encoding raw motion and spatial structure. With the flourishing of machine learning, big tech com-

panies have been developing large-scale models with billions or trillions of parameters. These large-scale models learn powerful representations, closer to human concepts and semantic.

A very good example is Audio-Visual HuBERT. AV HuBERT is a transformer trained to carry out lip-reading and AV speech recognition. It is trained upon findings from [Devlin et al., 2018] where the authors present a way of obtaining meaningful pre-trained representations for natural language, which led to Bidirectional Encoder Representations from Transformers (BERT). These representations are obtained by a self supervised training technique called "masked language modeling," in which a portion of the input text is randomly masked. The model is trained to predict the masked tokens based on the context provided by the remaining input text. The model is later fine-tuned for a wide range of downstream tasks. This training technique was extended to the audio domain in [Hsu et al., 2021]. In that work, the authors run a clustering algorithm on the input acoustic features to generate frame-level assignments (one per sequence element). Then, the model is trained to predict those, also applying the masking strategy from BERT, which forces the model to learn good acoustic features.

AV HuBERT extends this strategy to multimodal data similarly. They incorporate a modality dropout, in which one modality is disabled randomly to force the model to solve the task from both modalities. This pre-training strategy and the expressiveness (large amount of parameters) of the network results in very powerful visual and acoustic features, which encode high-level information with semantic meaning.

2.4 Foundations on Sound Source separation

2.4.1 Problem formulation

Blind Source Separation (BSS) is the problem of separating a set of mixed signals into their individual components, without any prior knowledge about the underlying sources or the mixing process. This is in contrast to informed source separation, where additional information about the

sources or the mixing process is leveraged to make the separation process more effective.

In the context of this thesis, the type of signals to isolate are voices and instruments, from which video information is always known. Hence, we are going to bound the problem to music source separation, speech separation, and singing voice separation. While acoustic characteristics are different among them, the way of tackling the separation process is the same when using machine learning. We also assume mono audio, which is predominant in video-streaming platforms.

Given a set of N acoustic signals $\{x_i(t)|i = 1, ..., N\}$, we can model a mixture of signals as their linear combination:

$$x_m(t) = \sum_{i=1}^{N} x_i(t).$$

This is an approximation. In music industry, tracks are mixed with high non-linearity, e.g., when filtering. There exist also natural phenomena such as reverberation that violate this assumption.

Our goal is to recover each independent signal $x_i(t)$. This problem is undetermined, as there exists a single observation of the mixture. This problem can be solved with different strategies. In general, when using deep learning, a typical approach is feeding a NN with the mixture and, optionally, other kinds of information that guides the source separation process (one-hot categorical vectors[Slizovskaia et al., 2021], video, text [Rahimi et al., 2022]...).

We can classify the NNs depending on whether they work directly on waveforms, their time-frequency representation, i.e. their STFT, or both (hybrid approaches). Most of the classical algorithms for source separation used to work in the time-frequency domain. Besides, computer vision has traditionally introduced the most advanced architectures in deep learning. As STFT is a 2D signal, those could be easily adapted. There exist some technical reasons as well. Time-frequency representations are more disentangled than time-domain ones. In addition, time-frequency source separation was carried out with a masking system, which is simpler (but less powerful) to estimate than a whole signal directly.

2.4.2 Sound Source Separation in the time-frequency domain

Recalling the notation from Sec. 2.1.2, let $x[t]$ be a generic discrete time audio waveform and $X[k, l]$ be its STFT, where $k = 0, ..., K-1$ is a frequency index and $l = 0, ..., L-1$ is a frame index.

When working in the time-frequency domain, the usual approach is to predict a mask that acts as a filter by enhancing the target sources. The reasons to do so are two-fold: first, the isolated signals are already present in the mixture, therefore it is possible to isolate them rather than re-synthesize them. Second, energy is usually concentrated in the low-frequencies of the spectrum, as shown in Fig. 2.7. Hence, high frequencies would be underrated with Euclidean distances in case of predicting the magnitude STFT directly. Besides, harmonics and structured patterns are weaker in the high frequencies. On the contrary, mask energy is homogeneously distributed, and a gradient penalty based on the energy of the mixture can be applied. This way, the contribution of each time-frequency bin in the mask to the loss is proportional to the energy of the analogous bin in the mixture.

Masks have quickly evolved. There exist three main types: binary masks, ratio masks, and complex masks. In Eqs. 2.1- 2.4, M_i stands for the mask that isolates the i-th source.

Binary masks represent the presence or absence of a particular sound source in the mixture. Each time-frequency bin is set to either 0 or 1. A value of 0 indicates that the corresponding sound source is not present in the mixture, while a value of 1 indicates that it is present. Binary masks predicted by a NN are probability maps, since neural networks cannot generate binary numbers. Binary masks can be formulated in several ways. In this thesis, two types of binary masks are used: soft binary masks, defined in Eq. 2.1, where each time-frequency point can be assigned to several sources. And hard binary masks, defined in Eq. 2.2,

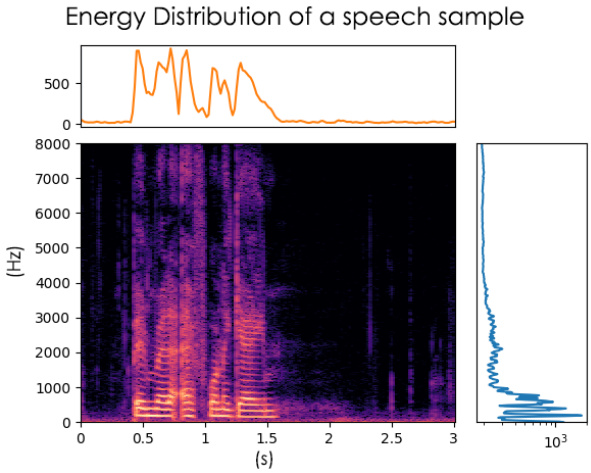

Figure 2.7: Energy distribution of a speech sample along frequency and time. Note that the energy distribution along frequency is log-scaled.

where each time-frequency point can be assigned only to a single source.

$$M_i[k,l] = \begin{cases} 1, & \text{if } \|X_i[k,l]\| \geq \|X_i[k,l] - X_m[k,l]\|, \\ 0, & \text{otherwise.} \end{cases} \tag{2.1}$$

$$M_i[k,l] = \begin{cases} 1, & \text{if } \|X_i[k,l]\| \geq \|X_n[k,l]\| \ \ \forall n \in \{1,...,N\}, \\ 0, & \text{otherwise.} \end{cases} \tag{2.2}$$

Ratio masks are another type of masks used in sound source separation. They indicate the relative strength or presence of the sound source in the mixture. Ratio masks are considered a "soft" masking, where the mask is used to attenuate or reduce the strength of a particular sound source in the mixture, rather than completely removing it. Ratio masks can be computed according to Eq. 2.3. Note that, due to the fact it is computed as a division between modules of complex numbers, the resulting values can be greater than 1. In practice, the mask is clipped,

$$M_i[k,l] = \frac{\|X_i[k,l]\|}{\|X_m[k,l]\|}. \tag{2.3}$$

Binary and ratio masks are only capable of modifying the magnitude of the spectrogram. In this case, the phase of the mixture is used to reconstruct the isolated waveform. Alternatively, the phase can be estimated with algorithms such as Local Weighted Sums [Le Roux et al., 2010], or GriffinLim [Griffin and Lim, 1984].

More recently, **complex masks** were developed. Unlike binary and ratio masks, which only modulate the magnitude, complex masks also modulate the phase. Phase quality is related to robotic effects and naturalness, and have proven to improve results in speech separation in works as [Ephrat et al., 2018] or [Williamson et al., 2015]. An implicit formulation of complex masks is shown in Eq. 2.4, where $\otimes$ denotes complex product. The magnitude of complex masks are unbounded as well. A common approach is using a hyperbolic tangent of real and imaginary parts [Williamson et al., 2015]. This bounds the values and stabilizes gradients,

$$M_i[k,l] \otimes X_m[k,l] = X_i[k,l]. \tag{2.4}$$

2.4.3 Sound Source Separation in the time domain and hybrid approaches

Generally speaking, estimating the phase of a spectrogram is not straightforward. That is why, concurrently to complex masks, time-domain sound source separation gained attention. The first competitive attempt proposes a 1D U-Net for sound source separation [Stoller et al., 2018]. This approach is data greedy, as 1D CNNs tend to have more parameters due to their large kernels. In addition, the isolated sources were estimated directly, mainly via mean square error or absolute error. This is harder than estimating masks, though potentially more powerful, and incurs the well-known problematic of average-smoothing derived from using Euclidean distances.

Time-frequency domain is more disentangled than time domain. This has very recently lead to hybrid approaches, where NNs are exposed to both, waveforms and STFT in parallel [Rouard et al., 2022].

2.4.4 Sound Source Separation before Deep Learning

While this thesis is focused on deep learning approaches, BSS was already tackled with classical algorithms. Traditionally, BSS in the monoaural case (a single microphone) has been approached with matrix decomposition algorithms such as Independent Component Analysis (ICA), e.g. in [Hyvärinen and Oja, 2000], concurrently to ICA, with sparse decomposition [Zibulevsky and Pearlmutter, 2001], Non-negative Matrix Factorization (NMF) [Virtanen, 2007], Computational Auditory Scene Analysis (CASA) [Ellis, 1996], or Probabilistic Latent Component Analysis (PLCA) [Smaragdis et al., 2006].

2.4.5 Mix-and-separate Strategy

High-performance, robust ML methods require large amounts of data covering all possible scenarios. If this condition is accomplished, and models are expressive enough, they can generalize pretty well. One of the challenges introduced in Sec. 1.2.2, was related to the difficulty of labeling audio data. In real-world acoustic performances, sources are interlaced. Onsets usually happen at the same time, there is reverberation, chorals, similar tempo etcetera... Collecting real-world ground-truth data implies the capability of recording isolated sources playing at the same time, which is very difficult. There are some attempts from researchers working in music source separation, where players or singers were recorded in different isolated chambers and synchronized using headphones [Li et al., 2019]. Even following this strategy, it is very expensive to collect large scale datasets.

To overcome the lack of labeled data, the *mix-and-separate* strategy is often used [Zhao et al., 2018]. Taking advantage of the linearity of the mixtures, an unlimited amount of scenarios can be generated synthetically by combining the sources smartly together with different acoustic resources that emulate real artifacts.

Part I

Audio-visual Source Separation

Chapter 3

INSTRUMENTAL MUSIC SOURCE SEPARATION

3.1 Introduction

There is a growing interest in multimodal techniques for solving Music Information Retrieval (MIR) problems. Music performances have a highly multimodal content and the different modalities involved are highly correlated: sounds are emitted by the motion of the player performing and in chamber music performances the scores constitute an additional encoding that may be as well leveraged for the automatic analysis of music [Li et al., 2017a].

A fundamental problem in audio processing is Blind Source Separation, which was already introduced in Sec. 2.4. As a quick reminder, BSS consists in, given a mixture of signals, recovering the individual signals the mixture is conformed by. In speech, it is also known as the Cocktail Party problem, which refers to the task of recognizing an individual speech in noisy social environments [Cherry, 1953].

If the mixture recording provides as well the visual information, as it is the case for videos, the additional modality can be also processed to help in the source separation task. Indeed, by visually inspecting the scene we may extract information about the number of sound sources, their type, spatio-temporal location and also motion, which naturally correlates with the emitted sound. We can find pioneering works that make use of audio-visual data for sound localization [Hershey and Movellan, 2000, Kidron et al., 2005]. In the context of music, visual information has also proven to help model-based methods in source separation and localization [Li et al., 2017a, Li et al., 2017b, Parekh et al., 2017]. With the flourishing of deep learning, many recent works exploit both, audio and video content, to perform music source separation [Gao and Grauman, 2019, Zhao et al., 2019, Xu et al., 2019], source localization, as proposed in [Arandjelović and Zisserman, 2018] or both at the same time, in papers such as [Zhao et al., 2018, Zhu and Rahtu, 2021, Zhu and Rahtu, 2020]. These works use networks that have been trained in a self-supervised way using pairs of corresponding/non-corresponding audio-visual signals for localization purposes or the *mix-and-separate* approach for source separation [Zhao et al., 2018, Gao and Grauman, 2019, Zhao et al., 2019,

Xu et al., 2019]. In this chapter, artificial mixtures are created in the *mix-and-separate* strategy by combining individual sources of the dataset, as explained in Sec. 2.4.5. This chapter corresponds to the following publication:

"Solos: A Dataset for Audio-Visual Music Source Separation and Localization" *J.F. Montesinos*, O. Slizovskaia, G. Haro. In *In 22st IEEE International Workshop on Multimedia Signal Processing, MMSP 2020.*

This chapter presents *Solos*, a new dataset of music performance recordings of soloists that can be used to train deep neural networks, using the mix-and-separate strategy, both for source separation and sound localization problems. Compared to a similar dataset of music instruments presented in [Zhao et al., 2018] and its extended version [Zhou et al., 2019], our dataset does contain the same type of chamber orchestra instruments present in the University of Rochester Multi-Modal Music Performance Dataset (URMP) dataset [Li et al., 2019]. Solos is a dataset of 755 real-world recordings gathered from YouTube which provides several features missing in the aforementioned datasets: hands position ground-truth and manually-curated timestamps. Source localization is usually indirectly learned by networks. Thus, providing a practical localization ground-truth is not straightforward. Nevertheless, networks often point to the player hands as if they were the sound source. We expect hands localization can help to provide additional cue to improve audiovisual source separation, or can be used as source ground-truth localization.

3.2 Related Work

The rising amount of works in audio-visual analysis reflects the importance of having new and better datasets.

URMP [Li et al., 2019] is a dataset with 44 multi-instrument video recordings of classical music pieces. Each instrument present in a piece was recorded separately, both with video and high-quality audio with a stand-alone microphone, in order to have ground-truth individual tracks.

Although playing separately, the instruments were coordinated by using a conducting video with a pianist playing in order to set the common timing for the different players. After synchronization, the audio of the individual videos was replaced by the high-quality audio of the microphone and then different recordings were assembled to create the mixture: the individual high-quality audio recordings were added up to create the audio mixture and the visual content was composited in a single video with a common background where all players were arranged at the same level from left to right. For each piece, the dataset provides the musical score in MIDI format, the high-quality individual instrument audio recordings and the videos of the assembled pieces. The instruments present in the dataset, shown in Figure 3.1, are common instruments in chamber orchestras. In spite of all its good characteristics, it is a small dataset and thus not appropriate for training deep learning architectures.

Shortly before, two other datasets of audio-visual recordings of music instruments performances have been presented: MUSIC [Zhao et al., 2018] and MusicES [Zhou et al., 2019]. MUSIC consists of 536 recordings of solos and 149 videos of duets across 11 categories: accordion, acoustic guitar, cello, clarinet, erhu, flute, saxophone, trumpet, tuba, violin and xylophone. This dataset was gathered by querying YouTube. MusicES [Zhou et al., 2019] is an extension of MUSIC to around the triple of its original size with approximately 1475 recordings but spread in 9 categories instead: accordion, guitar, cello, flute, saxophone, trumpet, tuba, violin and xylophone. There are 7 common categories in MUSIC and Solos: violin, cello, flute, clarinet, saxophone, trumpet and tuba. The common categories between MusicES and Solos are 6 (the former ones except clarinet). Solos and MusicES are complementary. There is only an small intersection of 5% between both, which means both datasets can be combined into a bigger one.

We can find in the literature several examples which show the utility of audio-visual datasets. *The Sound of Pixels* [Zhao et al., 2018] performs audio source separation generating audio spectral components which are further smartly selected by using visual features coming from the video stream to obtain separated sources. This idea was further extended in

[Xu et al., 2019] in order to separate the different sounds present in the mixture in a recursive way. At each stage, the system separates the most salient source from the ones remaining in the mixture. *The Sound of Motions* [Zhao et al., 2019] uses dense trajectories obtained from optical flow to condition audio source separation, being able even to separate same-instrument mixtures. Visual conditioning is also used to separate different instruments [Gao and Grauman, 2019]; during training, a classification loss is used on the separated sounds to enforce object consistency and a co-separation loss forces the estimated individual sounds to produce the original mixtures once reassembled. A cascade strategy is proposed in [Zhu and Rahtu, 2020]. The idea is to carry out motion-based AV source separation and use the motion from the remaining sources (i.e. the non-target ones) to filter out interferences. The same authors proposed a more complex separation framework in [Zhu and Rahtu, 2021]. First, appearance-based source separation is done, to, later on, be refined using motion.

In [Parekh et al., 2017], the authors developed an energy-based method which minimizes a NMF term which is forced to be aligned to a matrix containing per-source motion information. This motion matrix contains the average magnitude velocities of the clustered motion trajectories in each player bounding box.

Some works show the rising use of skeletons in audiovisual tasks. In *Audio to body dynamics* [Shlizerman et al., 2017] authors show it is possible to predict skeletons reproducing the movements of players playing instruments such as piano or violin. Skeletons have proven to be useful for establishing audio-visual correspondences, such as body or finger motion with note onsets or pitch fluctuations, in chamber music performances [Li et al., 2019]. A recent work [Gan et al., 2020] tackles the source separation problem in a similar to *Sound of Motions* [Zhao et al., 2019] but replacing the dense trajectories by skeleton information.

Figure 3.1: Solos and URMP instrument categories. Image adapted from [Li et al., 2019].

3.3 Dataset

Solos[1] was constructed aiming to have the same categories as the URMP [Li et al., 2019] dataset, so that URMP can be used as testing dataset in a real-world scenario. This way we aim to establish a standard way of evaluating the performance of source separation algorithms, avoiding the use of *mix-and-separate* in testing. Solos consists of 755 recordings distributed amongst 13 categories as shown in Figure 3.1, with an average amount of 58 recordings per category and an average duration of 5:16 min. It is interesting to highlight that, for 8 out of 13 categories, the median of resolution is HD, despite being a YouTube-gathered dataset. Per-category statistics can be found in Table 3.1. These recordings were gathered by querying YouTube using the tags solo and auditions in several languages such as English, Spanish, French, Italian, Chinese or Russian.

3.3.1 OpenPose Skeletons

Solos is not only a set of recordings. Apart from the videos identificators We also provide: i) body and hand skeletons estimated by OpenPose [Cao et al., 2019] in each frame of each recording and ii) timestamps indicating useful parts. OpenPose is a system capable to predict body skeleton and hands skeletons making use of two different neural networks. To do

[1]Dataset available at https://github.com/JuanFMontesinos/Solos

Category	# Recordings	Mean duration	Median resolution
Violin	66	6:16	1080×720
Viola	55	5:31	1280×720
Cello	134	7:21	640 ×480
DoubleBass	58	8:53	1280×720
Flute	48	4:00	640×360
Oboe	53	5:45	1280×720
Clarinet	49	3:23	640 ×360
Bassoon	56	5:08	1280×720
Saxophone	45	2:42	1280×720
Trumpet	50	1:14	640×360
Horn	50	5:11	1280×720
Trombone	50	5:03	1280×720
Tuba	41	2:49	640×360
TOTAL	**755**	**5:16**	**854×480**

Table 3.1: Statistics of Solos Dataset

so, they predict a confidence map of the belief that a specific body part may be located at any given pixel as well as part affinity fields which encode the degree of association between different body parts. Finally, it predicts 2D skeletons and per-joint confidence via greedy inference. In practice, the body skeleton is estimated with a first network. Then, the position of the wrists in the body skeleton are used to estimate the position of both hands. A second neural network obtains the skeleton of each hand independently. Note that since each body part is estimated independently, OpenPose makes no assumptions about the limbs to find. It just calculates the most likely skeleton given confidence maps and part affinity fields. The whole process is carried out frame-wise. This leads to a small flickering and mispredictions between frames.

3.3.2 Timestamps estimation and skeleton refinement

Video streams are re-sampled to 25 FPS keeping the audio stream intact. An iterative process returns stamps for which there are at least N frames with a detected hand and no more than M consecutive mispredictions. In practice we use N=150 and M=5, thus, a minimum of 6 seconds of video with at most 5 consecutive frames with hand mispredictions. At this point, we have segments of video in which there are hands detected. To refine these results we further applied an energy-based silence detector which allows to discard those segments in which the instrument is not being played, e.g., transitions, music sheet changes, etcetera. Besides, we perform a linear interpolation of the mispredicted keypoints in a relative base of coordinates. Directly interpolating the absolute coordinates would lead to deformations of the skeleton and inaccuracies. Since skeletons are tree-like graphs it is possible to interpolate the relative coordinates of each joint (node in the graph) with respect to its parent node. Then, the absolute coordinates of the joint are recovered with the sum of the absolute coordinates of its parent and the estimated relative coordinates with respect to the parent. Let us denote by J_i^t the relative coordinates of the i-th joint with respect to its parent at time t. On the other hand, $\hat{J}_i^t$ denotes the estimated value of J_i^t when the i-th joint is mispredicted. $\hat{J}_i^t$ can be linearly interpolated using the relative coordinates of the closest i-th detected joint before time t (i.e $J_i^{t^-}$ where $t^- < t$), and analogously with the closest i-th detected joint after time t (i.e $J_i^{t^+}$ where $t < t^+$). For example, given the following sequence of detected and misdetected coordinates (that need to be estimated), J and $\hat{J}$ respectively:

$$\{J_i^{t-n}, \hat{J}_i^{t-n+1}, ..., \hat{J}_i^t, ..., \hat{J}_i^{t+m-1}, J_i^{t+m}\}$$

then, the interpolation at time t can be calculated as:

$$\hat{J}_i^t = \frac{m\, J_i^{t-n} + n\, J_i^{t+m}}{m+n}.$$

OpenPose maps mispredicted joints to the origin of coordinates. We empirically found that such a big jump in the position of a joint induces noise. Using interpolated coordinates helps to address this problem.

3.4 Experiments

In order to show the suitability of Solos, we have trained *The Sound of Pixels* (SoP) [Zhao et al., 2018] and Multi-head U-Net (MHU-Net), proposed [Doire and Okubadejo, 2019]. We have carried out four experiments: i) we have evaluated the SoP pre-trained model provided by the authors, ii) we have trained SoP from scratch, iii) we have fine-tuned the pre-trained network in our dataset and iv) we have trained Multi-head U-Net from scratch. MHU-Net has been trained to separate mixtures with the number of sources varied from two to seven following a curriculum learning procedure as it improves the results. SoP has been trained according to the optimal strategy described in [Zhao et al., 2018].

Evaluation is performed on the URMP dataset [Li et al., 2019] using the real mixtures they provide. URMP tracks are sequentially split in 6s-duration segments. Metrics are obtained from all the resulting splits.

3.4.1 Architectures and training details

We have chosen *The Sound of Pixels* as baseline since its weights are publicly available and the network is trained in a straight-forward way. SoP is composed of three main sub-networks: A dilated ResNet [Yu et al., 2017] as video-analysis network, a variant of U-Net [Ronneberger et al., 2015] as audio-processing network and an audio synthesizer network. We also compare its results against a MHU-Net [Doire and Okubadejo, 2019].

U-Net [Ronneberger et al., 2015] is an encoder-decoder architecture with skip connections in between, as described in Sec. 2.2.1. Skip connections help to recover the original spatial structure. MHU-Net is a step forward as it consist of as many decoders as possible sources. Each decoder is specialized in a single source improving performance.

The Sound of Pixels (SoP) [Zhao et al., 2018] does not follow the original U-Net architecture, but the U-Net described in [Jansson et al., 2017], which was tuned for singing voice separation. Instead of having two con-

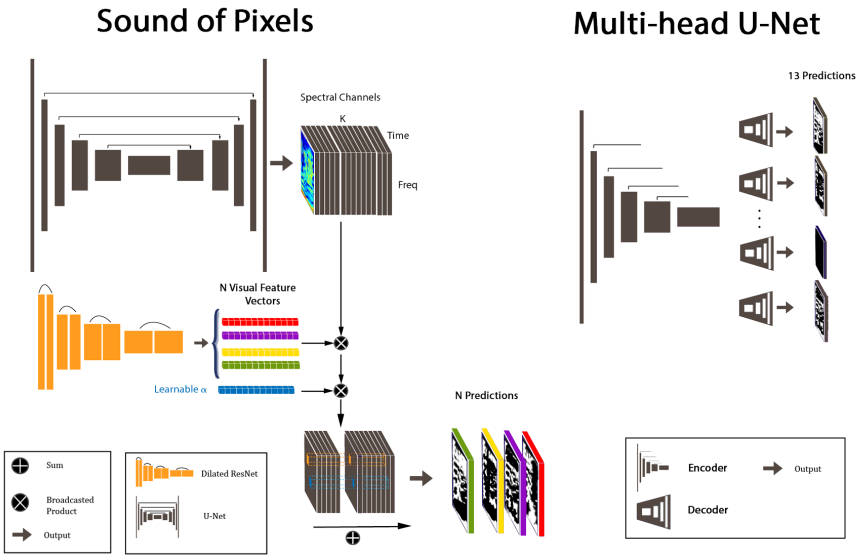

Figure 3.2: Considered architectures. Left, Sound of Pixels: The network takes as input a mixture spectrogram and returns a binary mask given the visual feature vector of the desired source. Right, Multi-Head U-Net: It takes as input a mixture spectrogram and returns 13 ratio masks, one per decoder.

volutions per block followed by max-pooling, they use a single convolution with a bigger kernel and striding. The original work proposes a central block with learnable parameters whereas the central block is a static latent space in *SoP*. U-Net has been widely used as backbone of several architectures for tasks such us image generation [Liu et al., 2018], noise suppression and super-resolution [Mao et al., 2016], image-to-image translation [Isola et al., 2017], image segmentation [Ronneberger et al., 2015] or singing voice separation [Jansson et al., 2017]. SoP U-Net consists of 7 blocks with 32, 64, 128, 256, 512, 512 and 512 channels respectively (6 blocks for the MHU-Net). The latent space can be considered as the last output of the encoder. Dilated ResNet is a ResNet-like architecture which makes use of dilated convolutions to keep the receptive field while

increasing the resulting spatial resolution. The output of the U-Net is a set of 32 spectral components (channels) which are the same size than the input spectrogram, in case of SoP, and a single source per decoder in case of MHU-Net. Given a representative frame, visual features are obtained using the Dilated ResNet. These visual features are nothing but a vector of 32 elements (which corresponds to the number of output channels of U-Net) which are used to select proper spectral components. This selection is performed by the audio analysis network which consist of 32 learnable parameters, α_p, plus a bias, β. This operation can be mathematically described as follows:

$$\beta + \sum_{p=1}^{32} \alpha_p v_{f_p} S_p[k, l],$$

where $S_p[k, l]$ is the p-th predicted spectral component at frequency-time bin $[k, l]$.

Figure 3.2 illustrates the SoP configuration. It is interesting to highlight that making the visual network to select the spectral components forces it to indirectly learn instrument localization, which can be inferred via activation maps.

On one hand, MHU-Net has been trained using a curriculum learning strategy that consists of a gradual increment on the amount of sources present in the mixture from two to four. When the loss stays on a plateau for more than 160,000 iterations, the amount of sources is increased by one. We have used mean-square error loss, ADAM optimizer (proposed in [Kingma and Ba, 2014]), an initial learning rate (LR) of 10^{-4}, weight decay of 10^{-5} and dropout of 0.2 in the decoder. We have also reduced the LR by a half if the loss stays on a plateau for more than 400,000 iterations.

On the other hand, SoP has been trained using a LR of 10^{-3} for the U-Net and a LR of 10^{-4} for the Dilated ResNet as it was pre-trained on ImageNet. We have applied a weight on the gradients based on the magnitude of the mixture spectrogram so that time-frequency points of the predicted source/s contribute to the loss according to the energy of the analogous time-frequency points in the mixture spectrogram, as mentioned in Sec.

2.4. We used different training strategies for SoP and MHU-Net as the optimal training for SoP harms the performance of the MHU-Net.

3.4.2 Data pre-processing

In order to train the aforementioned architectures, audio is re-sampled to 11025 Hz and 16 bit. Samples fed into the network are 6s duration. We use STFT to obtain time-frequency representations of waveforms. STFT is computed using Hanning window of length 1022 and hop length 256 so that we obtain a spectrogram of size 512×256 for a 6s sample. Later on, we apply a log re-scale on the frequency axis expanding lower frequencies and compressing higher ones. Lastly, we convert magnitude spectrograms into dB w.r.t. the minimum value of each spectrogram and normalize between -1 and 1. For training SoP we have used hard binary masks as ground-truth masks, as described in Sec. 2.4.2, while for MHU-Net ratio masks were used.

3.4.3 Mix-and-separate

In Section 2.4.5, the mix-and-separate strategy was introduced. The idea is generating artificial mixtures by combining individual isolated sources. Standard floating-point audio format imposes a waveform to be bounded between [-1,1]. At the time of creating artificial mixtures resulting waveforms may be out of these bounds. This can help neural networks to find shortcuts to overfit. To avoid this behavior spectrograms are clamped according to the equivalent bounds in the time-frequency domain.

Recalling the notation from Sec. 2.1.2, STFT can be computed as:

$$X[k,l] = \sum_{m=0}^{M-1} g[m]x[m+lL]e^{-j2\pi mk/K}$$

Since $x[t] \in [-1, 1]$ it can be easily shown that:

$$\|X[k, l]\| \leq \sum_{n=0}^{M-1} \|g[m]\|,$$

i.e., the magnitude STFT of an audio signal bounded between [-1,1] is bounded between $[0, \sum |w[k]|]$. Thus, given the mixture resulting of N waveforms, the spectrogram of the mixture is defined the following way:

$$X_{mix}[k, l] = \min \left(\sum_{n=1}^{N} X_n[k, l], \sum |g[m]| \right),$$

which is equivalent to:

$$X_{mix}[k, l] = STFT \left\{ \min \left(1, \max \left(\sum_{n=1}^{N} x_n(t), -1 \right) \right) \right\}.$$

3.4.4 Results

Benchmark results for Source to Distortion Ratio (SDR), Source to Interferences Ratio (SIR), and Sources to Artifacts Ratio (SAR) proposed in [Vincent et al., 2006] are shown in Table 3.2 in terms of mean and standard deviation. As it can be observed, SoP evaluated using its original weights performs the worst. One possible reason for that could be the absence of some of the URMP categories on the MUSIC dataset. If we train the network from scratch on Solos, results improve by almost 1 dB. However, it is possible to achieve an even better result fine-tuning the network, pre-trained with MUSIC, on Solos. We hypothesize that the improvement occurs as the network is exposed to much more training data. Moreover, the table results show how it is possible to reach higher performance by using more powerful architectures like MHU-Net.

	SDR ↑	SIR ↑	SAR ↑
SoP [Zhao et al., 2018]	-3.76 ± 4.00	-1.45 ± 4.68	7.56 ± 3.13
SoP-Solos	-2.98 ± 5.07	0.46 ± 6.76	6.37 ± 2.94
SoP-ft	-2.57 ± 4.99	0.47 ± 6.43	6.89 ± 2.48
MHU-Net	-0.56 ± 5.96	1.04 ± 7.24	10.37 ± 3.48

Table 3.2: Benchmark results. SoP original weights, SoP-Solos: Sound of Pixels trained from scratch on Solos. SoP-ft: Sound of Pixels finetuned on Solos. MHU-Net: Multi-head U-Net with 13 decoders.

3.5 Conclusions

We have presented Solos, a new audio-visual dataset of music recordings of soloists, suitable for training source separation deep neural networks using the mix-and-separate strategy for self-supervised learning. There are 13 different instruments in the dataset; those are common instruments in chamber orchestras and the ones included in the University of Rochester Multi-Modal Music Performance dataset [Li et al., 2019]. The characteristics of URMP – small dataset of real performances with ground truth individual stems – make it a suitable dataset for testing purposes but as far as we know, so far there is no existing large-scale dataset with the same instruments as in URMP. Two different networks based on the U-Net architecture have been trained in the new dataset and further evaluated in URMP, showing the impact of training on the same set of instruments as the test set. Moreover, Solos provides hands localization and timestamps to video intervals where hands are sufficiently visible. This information could be useful for training purposes and also for learning to solve the task of sound localization.

Chapter 4

CNN-BASED SINGING VOICE SEPARATION

4.1 Introduction

Voices form an integral part of our daily lives. In the form of speech, human voice serves as an effective means of communication. The same voice, when vocalised in sustained tonality and/or rhythm, turns into something musical: the singing voice. The singing voice has become a vital element in the music industry today. Apart from its usage as lead singing voice in songs, it is also found in other diverse forms like rap music, opera singing, solfege, scatting, humming, backing vocals and beatboxing to name a few. *A cappella* refers to a musical arrangement with single or multiple singing voices without any instrumental accompaniment. We are interested in isolating the target voices of interest in multi-voice *a cappella* videos, and in general, in music videos with singing faces.

Singing voice separation has been largely explored in the context of separating voice from instrumental accompaniment. The audio-only models developed for separating the singing voice from the instrumental accompaniment (e.g. works like [Takahashi et al., 2018, Samuel et al., 2020, Li et al., 2021b]) largely benefit from the differences in the timbral characteristics between the singing voice and the accompaniment. However, such models do not perform well in the case of separating a particular voice from a mixture of overlapping voices or when the volume of the desired target voice is low. In fact, a very similar problem appears in speech separation when there are overlapping speech segments from different sources in a speech mixture. Audio-visual speech separation methods leverage visual information to isolate the desired target speech, and have been shown to outperform their audio-only counterparts as will be later explained in Chapter 5.

On contrary, AV singing voice separation methods remain unexplored in the literature. A system capable of isolating the target voice of interest in an audio mix has many applications. Such a system could be helpful in evaluating individual singing voices in multi-voice audio mixtures. It can also be useful for automatic karaoke generation, music unmixing and remixing, lyrics and pitch transcription, pitch correction and melodic analysis. Therefore, it is of our interest to improve upon the audio-only

singing voice separation method by incorporating visual information. Besides, whereas there are different AV benchmark datasets for speech separation (reviewed in [Michelsanti et al., 2020]), to the best of our knowledge, to date, there is no public dataset available for AV singing voice. This chapter is devoted to overcoming these issues, and relies on the following publication:

"A cappella: Audio-visual Singing Voice Separation" *J.F. Montesinos*, V.S. Kadandale, G. Haro. In *32nd British Machine Vision Conference, BMVC 2021*

Through the chapter, we show that using the visual features is particularly advantageous in the singing voice separation task, especially in the aforementioned challenging cases: multi-voice mixtures and mixtures with low volume target singing voice, which is coherent to AV speech separation. To this extent, and to alleviate the lack of data, we contribute to singing voice separation in the following terms:

i We created *Acappella*, a new dataset of solo singers performing with no accompaniment. This dataset can be used to train audio-visual networks for singing voice separation or for style/voice conversion.

ii A new AV deep neural network for singing voice separation that uses a spatio-temporal graph convolutional network to extract motion features from face landmarks efficiently.

iii An ablation of four different possibilities for the visual network in the audio-visual architecture and two different training settings.

iv An ablation on the performance of the network mentioned in ii), showing our proposal generalizes pretty well for unseen-unheard languages and singers.

The proposed architecture is based on a U-Net that processes a complex spectrogram and is conditioned by the motion features extracted by a spatio-temporal graph convolutional network that receives a sequence of face landmarks.

Although there are recent works that use graph neural networks with face landmarks for face identification [Papadopoulos et al., 2021] and emotion recognition [Ngoc et al., 2020], or with skeletons for separating musical instruments [Gan et al., 2020], to our knowledge, we are the first ones to use face landmarks processed with a graph neural network for audio-visual source separation in the speech/singing context. The U-Net architecture has been extensively used both in audio-only source separation methods [Meseguer-Brocal and Peeters, 2019, Jansson et al., 2017, Kadandale et al., 2020, Stoller et al., 2018], as well as in its AV counterpart [Gao and Grauman, 2019, Owens and Efros, 2018, Zhao et al., 2018, Zhao et al., 2019, Xu et al., 2019, Zhu and Rahtu, 2020]. We can also find works on source separation that condition the U-Net on prior information such as the presence of certain types of musical instruments, such as the research line of [Slizovskaia et al., 2019, Slizovskaia et al., 2021], also conditioning on phoneme activation for singing voice source separation [Meseguer-Brocal and Peeters, 2020] or the fundamental frequency of each type of voice sources in choir ensembles [Petermann et al., 2020].

Both the dataset and our model are, to the best of our knowledge, the first ones presented in the literature for audio-visual singing voice separation with publicly available code and data for reproducibility.

4.2 Related work

In the audio-visual speech separation works, there are multiple ways in which the visual features are extracted, depending on the front-end representation of the visual information. Many of such works [Wu et al., 2019, Nguyen et al., 2020, Li and Qian, 2020, Gabbay et al., 2018] operate directly on the mouth region of the video input to extract the lip motion features. In [Morrone et al., 2019], the motion vectors of face landmarks are used as input to an LSTM-based network. On the other hand, pretrained face embeddings are used in [Ephrat et al., 2018]. These embeddings are extracted as in [Cole et al., 2017], using the input video frames containing the whole face. They are invariant to illumination, pose, and

facial expression. The authors show that, apart from the region around the mouth, the facial parts like eyes and cheeks also contribute to the speech separation performance. A very recent work [Gao and Grauman, 2021] leverages not only the lip motion features but also the facial appearance of the speaker since it is related to certain speech attributes. Their network is trained in a multi-task fashion that jointly learns audio-visual speech separation and cross-modal face-voice embeddings that assist in establishing face-voice mappings. In [Chung et al., 2020a], a single face image of the target speaker is used to condition an audio-visual source separation model based on facial appearance. The correlation of voice traits and facial attributes has also proven useful in speaker identification [Kim et al., 2018] and image generation [Oh et al., 2019] tasks. Further, [Fernandez-Lopez et al., 2017] points out that facial expressions are helpful in the visual speech recognition task.

In a concurrent work, [Li et al., 2021a] explored the specific task of audio-visual singing voice separation. Li's audio-visual singing voice separation method particularly outperformed the audio-only baseline methods when the input sample contained backing vocals in addition to the target voice. Our work is along the similar lines but, in addition, we analyse the effect of volume of the target voice on the source separation quality. Further, our approach also differs from Li's work in terms of the choice of baseline models, the proposed model architecture, the experimental setup and the dataset.

4.3 The Dataset

In order to exploit the visual information in the singing voice separation problem, we gathered a new dataset of people singing *a cappella*, i.e. with no music accompaniment. The dataset, named *Acappella*, comprises around 46 hours of *a cappella* solo singing videos (i.e. a single singer per video) sourced from YouTube, sampled across different singers and languages. It covers four language categories: English, Spanish, Hindi and others.

The samples in our dataset are defined based on the timestamps corresponding to the segments of interest in each of the videos. These timestamps are provided in the dataset. They have been manually selected to exclude parts of the videos that do not satisfy any of the following characteristics: single frontal face view without occlusions, minimal background noise, no beatboxing, no snapping fingers, songs with lyrics.

Along with the dataset, we provide the splits for training set, validation set and test set. The training set makes up around 80% of the total dataset. Around 7% of the dataset forms the validation set which is used during the training to save the best checkpoint. The test set is divided into the following subsets: seen-heard and unseen-unheard. The former consists of samples from known singers, i.e. singers present in the training set but singing different songs. The latter contains singers who are not a part of the training set. The unseen-unheard test subset also contains samples from languages not heard in the training set. It presents an approximately uniform distribution of samples across language categories and gender. Extended statistics of the complete dataset are shown in Figure 4.1.

[Li et al., 2021a] created a similar dataset. It comprises of 491 solo singing voice YouTube videos and 65 recorded ones, which overall sum up to 12 hours. To our knowledge, the dataset presented in this paper is the biggest dataset of audio-visual solo singing voice.

We also wanted to test our models to separate voices in multi-voice videos where multiple singing faces are put together in a single view. Since such videos do not provide us with the individual voices for each face, it is not possible to quantitatively evaluate our models on them. Hence, we assembled a multi-voice video ourselves. The mixture contains six voices sung by the same person. The lead voice content is in English and Zulu, there is a voice emulating a flute, and the rest pair up and sing in unison most of the time in Zulu. Background accompaniment music is also included in this mixture to add to the complexity.

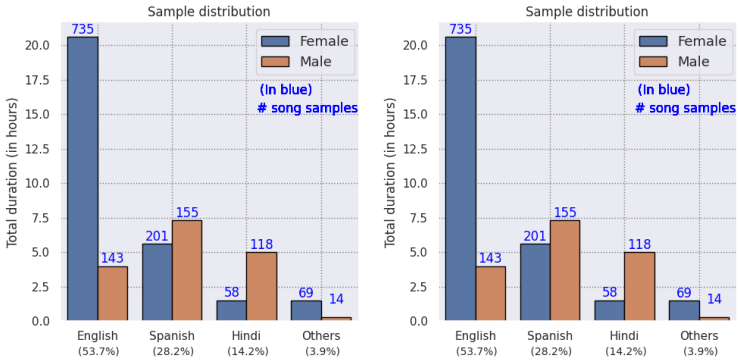

Figure 4.1: *Acappella* dataset statistics.

4.4 Singing voice separation model

Our model architecture comprises of a multimodal convolutional neural network which takes in a video and its corresponding mixture audio waveform and returns a complex mask (explained in Sec. 2.4). The waveform is mapped into the time-frequency domain using a STFT (see Sec. 2.1.2). The estimated mask allows recovering the separated voice of the target singer by computing the complex product between the mask and the spectrogram.

Our network is designed to receive only the visual information of the target singer to isolate, mainly for two reasons: i) it allows reducing and bounding the memory required for training, and ii) it broadens the applicability of the model since it only needs to be shown the face of the target singing voice with no additional visual information related to the other sources. This way, the model can handle mixtures of singing voice with accompaniments of different nature: musical instruments, backing vocals, other lead voices, beatboxing, snapping fingers, ambient sounds, or even different types of noise.

The architecture is a two-stream convolutional neural network for processing video and audio. It is denoted as Y-Net and illustrated in Figure 4.2. The audio network consists of a 6-blocks U-Net which predicts a

two-channel tensor. The U-Net [Ronneberger et al., 2015] is an encoder-decoder architecture with skip connections in between which allows to preserve the spatial structure while increasing the receptive field through blocks. We have experimented with two different number of blocks in the U-Net (see a comparison in Table 4.2) to ensure the best performance without overfitting. As explained in the previous chapter, the original U-Net design doubles the amount of channels each block while it downsamples the spatio-temporal resolution by two. In our U-Net with six blocks, we keep both the temporal resolution and the amount of channels, fixed, in the last blocks (i.e. the features are downsampled only along the frequency domain). The rationale behind this is not to lose much temporal resolution so that the features coming from the visual modality can be aligned to the audio ones and condition on those. We fix the temporal resolution of the U-Net bottleneck to 16 frames; this ensures that there are no out-of-synchronisation issues between both modalities and at the same time ensuring a fine enough temporal resolution for the separation task. On the other hand, a recent work [Lee et al., 2021] applies a synchronisation module between video and audio modalities but they deal with strong miss-alignments (up to 0.36s) which is not the case in our videos.

For the video network, we experiment with four different options:

1) **Y-Net-g:** This network extracts motion features from a sequence of aligned face landmarks. The definition of face landmarks as graphs was detailed in Sec. 2.3.1, whereas the specific pre-processing in this chapter will be explained in Sec. 4.4.1. Recalling the knowledge from Sec. 2.3.1, we treat the face landmarks as undirected graphs, where nodes encode the cartesian position of each landmark on the image. The reasons for using landmarks instead of raw frames are: i) a sequence of face landmarks contains motion information of the face, ii) appearance information and video background are removed, making the system less prone to overfitting, and iii) the computation and storage cost is much less compared to processing of the video frames. To make profit of the landmarks, we use a variation of the spatio-temporal graph convolutional neural network from [Yan et al., 2018], denoted as ST-GCN. Graph CNNs are a generalisation of traditional convolutions. Akin to the traditional convolutions, given a

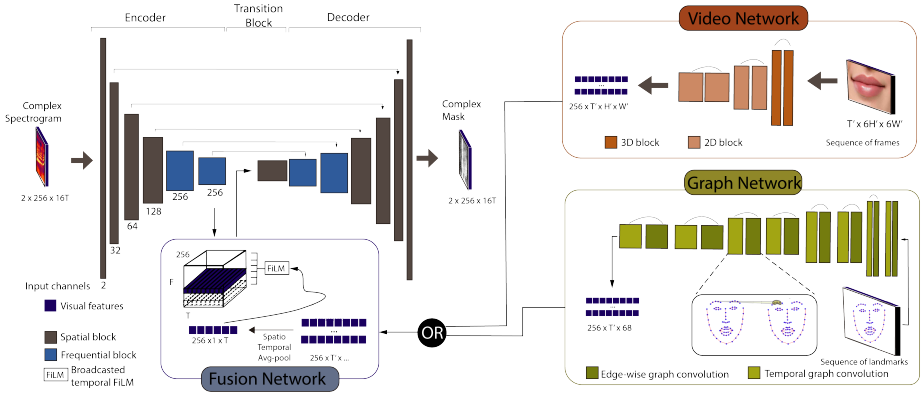

Figure 4.2: Y-Net model scheme. The system works with chunks of $4n$ seconds, where $n \in N$. The audio network takes as input a $256 \times 16Tn$ complex spectrogram and returns a complex mask. The visual network in case of Y-Net-m and Y-Net-mr, is the video network (in red), which takes as input a set of $100n$ frames cropped around the mouth of the target singer. In case of Y-Net-g and Y-Net-gr, the visual network is the graph network (in green) which takes as input a sequence of $68n$ landmarks of the face of the target singer. The visual features are fused with the audio network's latent space through a FiLM layer (we use $T = 16$). The FiLM broadcasts the $256 \times 1 \times T$ visual features into the $256 \times 16 \times T$ audio ones. The spatial blocks of the U-Net downsample in both, the frequency and the temporal dimension, while the frequential block downsamples along the frequency dimension only.

root node, ST-GCN works on a neighbour set of nodes as shown in Figure 4.2.

2) **Y-Net-m:** A sequence of video frames cropped around the mouth (more details in Section 4.4.1) are fed to a 3-block 3D-ResNet-like network, where the first block is 3D convolutional and the last two blocks are 2D convolutional. The 3D convolutional block processes motion information. This design turns into a network with 3M parameters (M stands for million). In contrast, a traditional 3D-ResNet18 has 33.4M and the 2D-

ResNet18 has 11.4M. This way, the visual network keeps the capacity to model spatio-temporal information, as suggested in [Tran et al., 2018], while having a contained amount of parameters not to overfit. A summary of the number of parameters of all models is shown in Table 4.1.

3) **Y-Net-e:** Inspired by works in AV speech separation , we consider the visual network used in [Ephrat et al., 2018]. The input to this visual network are the face embeddings extracted from the video frames cropped around the face, just like in [Ephrat et al., 2018]. The visual network comprises of six 1D dilated convolutional blocks.

We also experiment with the following additional configuration for the video network:

4) **Y-Net-f:** While the Y-Net-m ingests a sequence of video frames cropped around the mouth, Y-Net-f takes in a sequence of video frames cropped around the entire face. More details in Section 4.4.1.

Architecture	Visual Network	Audio-visual network
ResNet18	11.4M	-
3D-ResNet18	33.4M	-
LLCP	2.6M	15.3M
Y-Net-m	3.1M	11.4M
Y-Net-g	1.3M	9.4M

Table 4.1: Number of parameters (M for million) for the different architectures compared to common networks in computer vision (ResNet18 and 3D-ResNet18).

The visual features are fused with the audio networks' latent features via FiLM conditioning [Dumoulin et al., 2018]. Note that since both the audio and visual features are processed with convolutions, the time-frequency and spatio-temporal structures are kept, allowing to fuse them after an alignment in the temporal dimension. We apply a spatio-temporal average pooling to the video features to get the same number of features in the temporal dimension as the audio ones. At inference time, the model can work with chunks larger than 4s, only limited by the available memory, enabling a fast processing in contrast to processing chunks of 4s and

concatenating the resulting masks which could introduce artifacts.

4.4.1 Pre-processing

Video processing. Videos are resampled to 25 FPS to maintain uniform sampling rate across all the samples. We pre-processed the video stream of the target singer using a face detector[1] to extract 68 2D face landmarks, cropping around the face and aligning the face along all the frames in the video. In case of Y-Net-m, each frame is cropped around the mouth region and then resized to 96×96. Whereas, for Y-Net-f, each frame is cropped around the full face region and then resized to 128×96.Then, we feed the visual network with a sequence of 100 RGB frames, corresponding to 4s of video. In case of Y-Net-g, we feed the spatio-temporal graph network with the aligned sequence of face landmarks.
Audio processing. The audio signal is resampled to 16384 Hz. We consider a 4s-audio excerpt and compute its STFT using a Hanning window of size 1022 and a hop length of 256, the same way it is done in [Zhao et al., 2018, Gao and Grauman, 2019] which leads to a 512×256 spectrogram. This specific shape is useful to achieve a perfect alignment between the downconvolutional and the upconvolutional blocks of the U-Net, which are connected through the skip connections. For computational efficiency, we downsample the spectrogram in the frequency dimension and use a 256×256 spectrogram.

4.4.2 Training strategy, training target and loss

We train the networks in a self-supervised way by generating the audio mixtures artificially following the *mix-and-separate* strategy explained in Sec. 2.4.5. Given a set of N waveforms, $x_1, ..., x_N$, we generate an artificial mixture by taking the average, i.e. $x_m = \frac{1}{N} \sum x_i$. This way we can ensure the resulting mixture is bounded between -1 and 1. This differs from the previous chapter, where signals were clipped instead of aver-

[1] https://github.com/DinoMan/face-processor

aged. The network is trained to optimise an L_2 loss on bounded complex ratio masks [Williamson et al., 2015].

Let $X_i[k,l]$ be the STFT of a generic waveform x_i. Note that $X_i[k,l]$ is a complex matrix. We define the ideal complex ratio mask as follows: $M[k,l] = \frac{X_i[k,l]}{\sum X_i[k,l]}$.

Since the mask M is not bounded, we apply a hyperbolic tangent on the real and imaginary parts, M^r and M^i, respectively, to obtain a bounded complex mask:

$$M_b[k,l] = M_b[k,l]^r + M_b[k,l]^i j = \tanh(M^r[k,l]) + \tanh(M^i[k,l])\, j. \tag{4.1}$$

Let $\hat{M}_b$ be the bounded mask estimated by the network. The loss function is defined as:

$$\mathcal{L} = \|G^{\frac{1}{2}} \odot (\hat{M}_b^r - M_b^r)\|_2^2 + \|G^{\frac{1}{2}} \odot (\hat{M}_b^i - M_b^i)\|_2^2, \tag{4.2}$$

where $\odot$ denotes the element-wise product and G is a gradient penalty so that the points of the mixture spectrogram X_m with higher energy contribute more to the loss, it is defined as:

$$G[k,l] = \max(\min(\log(1 + \|X_m[k,l]\|), 10), 10^{-3}). \tag{4.3}$$

All the models have been trained using stochastic gradient descent, with a momentum of 0.8 and a weight decay of 10^{-5}. The learning rate is 0.01. Batch size of 10. In case of Y-Net-m, we use pretrained weights from Kinetics [Kay et al., 2017] and its statistics to normalise the input frames.

In Sec. 1.2.2, regarding the challenges of multimodal, we overviewed the problem of how to force the model to pay attention to one modality when the task is easy to solve from the other modality alone. When the patterns of each sound source are clearly different, the source separation is easier from the audio modality. To overcome this issue, we artificially create mixtures with different types of accompaniments, including human voices. By including human voices, the model must pay attention to the visuals in order to isolate the target voice. Since we only need the face of the target singer, we mix samples from *Acappella* together with samples from Audio Set [Gemmeke et al., 2017]. Audio Set is an in-the-wild

large-scale dataset of audio events across more than 600 categories. We gathered the categories related to the human voice and some typical accompaniments. These categories are: acappella, background music, beatboxing, choir, drum, lullaby, rapping, theremin, whistling and yodelling. We also include pop and rock music accompaniment from MUSDB18 dataset [Rafii et al., 2017]. While creating artificial mixtures, we ensure that all the samples from *Acappella* are used in each epoch. Those are mixed with a random sample from Audio Set or MUSDB18. We uniformly sample from all the accompaniment set categories. Including Audio Set in the training strategy increases the robustness of the model and addresses overfitting.

We consider different variants of our model: Y-Net-g, Y-Net-m, Y-Net-e and Y-Net-f. Note that these models, when referred to, without any additional suffix, indicate that they have been trained with mixtures that only contain one lead singing voice which is sourced from *Acappella* dataset and mixed with an accompaniment sample sourced from Audio Set or MUSDB18. On the other hand, we further append the suffix 'r' to the model name to indicate that it has been trained with mixtures in which, 50% of the time, the mixture contains an additional lead singing voice sourced from *Acappella* dataset. In this work, the experiments with model Y-Net-f are limited to its respective 'r' variant, Y-Net-fr, only.

4.5 Experiments

We conduct a set of experiments comparing the different Y-Net versions against their audio-only counterpart, the U-Net (i.e. our Y-Net without the visual network), and a state-of-the-art model for speech separation, the model of [Ephrat et al., 2018], that we denote as LLCP[2]. Results are expressed in terms of SDR and SIR, both defined in [Vincent et al., 2006].

We are interested in analysing the role of different types of visual information in different kind of mixtures. For that, we evaluate the models in two different setups: mixing a single singing voice with accompani-

[2]we use an existing code available at https://github.com/vitrioil/Speech-Separation

ment (one lead voice setup) and mixing two singing voices with accompaniment (two lead voices setup). Note that the singing voice(s) in both these setups are always sourced from *Acappella* dataset. Experiments are conducted both for seen-heard and unseen-unheard singers in heard languages and unseen-unheard singers in unheard languages (i.e. new languages) to check how the different networks generalise.

Models	4-blocks U-Net		6-blocks U-Net	
	SDR	SIR	SDR	SIR
U-Net	-1.92	12.16	-1.97	12.64
Y-Net-e	-1.50	12.50	–	–
Y-Net-m	**2.49**	14.04	**2.91**	**15.71**
Y-Net-g	1.85	**14.42**	2.07	15.49
Y-Net-fr	–	–	4.54	15.39
Y-Net-mr	3.38	13.81	5.03	15.80
Y-Net-gr	**4.71**	**15.67**	**6.41**	**17.38**

Table 4.2: Ablation study on the unseen-unheard test set in the two lead voices setup.

In Table 4.2, we show an ablation study of our model in the unseen-unheard test set in the two lead voices setup. We analyse four different aspects: i) the number of blocks in the U-Net, ii) audio-only versus audio-visual models, iii) the type of visual network, and iv) the training setting. First thing to notice is that the audio-only model, U-Net, performs much worse than the audio-visual ones and is the only model that does not get benefited from an increase of the number of blocks, since two lead voices are harder to separate from audio alone (actually, in the one lead voice setup U-Net does improve with more blocks). Thus, an increase in the U-Net blocks in the Y-Net models implies a gain in performance since the visual information is added to the network, which is indeed a crucial information to get a proper separation in the two lead voices setup. Second, both Y-Net-m and Y-Net-g perform better than Y-Net-e; from that, we hypothesise that visual embeddings do not sufficiently encode motion information. This follows the observations of [Cole et al., 2017], which explains that visual embeddings ignore factors of variation related to aspects

such as lighting, pose and expression (the latter being more related to the face motion). The Table 4.2 results also show how a boost in performance can be achieved if we train our models with mixtures in which 50% of the time two lead voices are present ('r' variants). This boost is particularly high (+2.86 dB and +4.34 dB in SDR, in 4-blocks and 6-blocks respectively) in the graph-based model, Y-Net-g, compared to the video-based model, Y-Net-m (+0.89 dB and +2.12 dB). Finally, we explore further (with the 'r' variants), which among the full face and the cropped mouth, works best as input to the video network; for that we include the results of the Y-Net-fr variant. Y-Net-fr relies on the same network as Y-Net-mr but the input to the video network are crops of the frames containing the full face rather than the lips region alone as in Y-Net-mr. Both the SDR and SIR values indicate that a better separation with the video network is achieved by limiting the visual information only to the lips region As it can be observed, the best model is Y-Net-gr with 6-blocks U-Net. From here on, all our model variants use a 6-blocks U-Net.

Model	Seen-Heard				Unseen-Unheard				Multi-voice
	English	Spanish	Hindi	Others	English	Spanish	Hindi	New Languages	English + Zulu
U-Net	-1.89	-2.25	-2.72	-1.42	-1.86	-2.34	-1.92	-2.15	5.18
LLCP	-0.55	-0.57	-1.08	-0.58	-0.9	-1.18	-0.73	-1.27	5.63
Y-Net-m	**4.17**	**3.60**	**3.50**	**3.19**	**3.28**	**3.33**	**2.11**	**2.31**	**7.24**
Y-Net-g	2.98	2.30	1.79	2.18	2.47	2.74	1.53	1.74	6.72
Y-Net-mr	7.78	5.42	5.32	5.82	5.33	5.14	4.35	4.07	6.51
Y-Net-gr	**8.61**	**6.62**	**5.91**	**7.45**	**6.73**	**6.72**	**5.76**	**5.27**	**7.21**

Table 4.3: SDR results in the two lead voices setup for different methods across languages, both in seen-heard and unseen-unheard test sets. SDR results also for the multi-voice case. LLCP stands for the work at [Ephrat et al., 2018].

Table 4.3 presents a comparison of our best models, Y-Net with video network or with graph network, with respect to the U-Net and LLCP models in both the seen-heard and unseen-unheard test sets in the two lead voices setup, as well as in the multi-voice recording with ground truth sources (singer not present in the training set). The SDR metrics are shown for different languages. We can observe that across models, the general tendency is that the performance increases for the languages more

represented in the training set. Again, U-Net performs the worst. LLCP outperforms U-Net but not our models. Models that have been trained with 50% of the samples containing two lead voices ('r' variants) have a boost in performance. Overall, it seems that the graph-based network can better exploit the motion information if the network is trained with the proper mixtures (two lead voices). The boost in performance with the Y-Net-gr is considerably and consistently higher than the boost with the Y-Net-mr, with an average boost of +1.89 dB for Y-Net-m and +3.98 dB for Y-Net-g.

In order to evaluate how sensitive are the different models to the volume of the target singer, we use different volume levels in the singing voice, so that experiments range from predominant singing voice to non-dominant one. To do so, each source X_i in the mixture is normalised by its root mean square value and then the singing voice is further multiplied by a factor α, where $\alpha \in \{0.25, 0.5, 1, 1.25\}$. Lastly, we rescale all the sources with the same value to ensure they are bounded between -1 and 1 while respecting the relative preset volumes (we divide by the maximum of absolute values of all sources). Figure 4.3 shows metrics for different volume levels and different methods in the unseen-unheard test set in two different setups: one lead voice (left) and two lead voices (right). For the one lead voice setup, LLCP performs the best in all volume levels. The second best for volume level factors of 1.25 and 1 is Y-Net-g, while for lower volume factors, 0.5 and 0.25, the second best is Y-Net-gr. Y-Net-gr exploits the motion information more than Y-Net-g and Y-Net-m since it has been trained with mixtures containing two lead voices, where motion is a key factor. This result shows that motion is also important in the case of one singing voice with a low volume, where the audio information alone is not enough to perform a good separation. Actually, we can observe how the In case of two lead voices, the Y-Net-gr is the best model for all volume levels (except for SIR in the lowest volume case, where it is the second best). The rest of our models are better than LLCP for volume levels of 1.25 and 1 (both in SDR and SIR) and better than LLCP in terms of SDR for volume level of 0.5. Overall, we can conclude that LLCP is a good choice for the one lead voice case and the Y-Net-gr is the

best model for two lead voices, where the motion features become crucial to get good separation results.

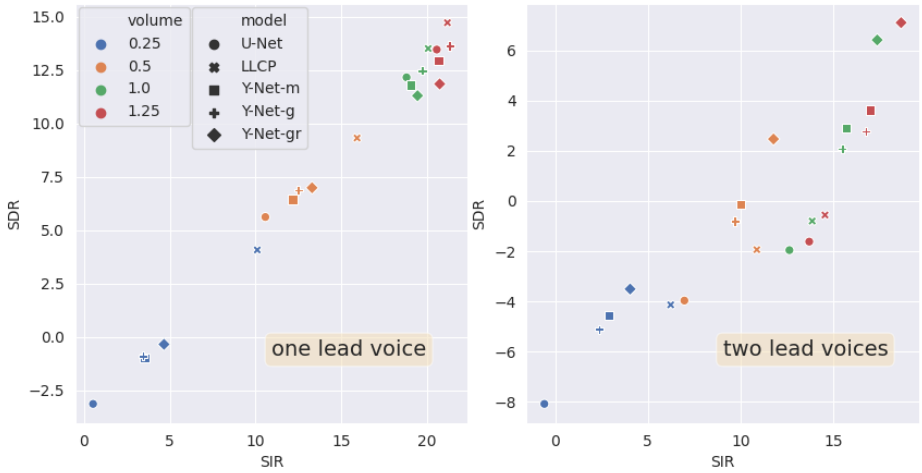

Figure 4.3: Results in the unseen-unheard test set: left, one lead voice setup; right, two lead voices setup. Different symbols are assigned to the different models and different colours to the different volume levels of the target voice.

We also compare the performance of the models based on the gender of the target voice and the non-target voice in the two lead voices setup in the unseen-unheard test subset. In the mixtures from this setup, there are four possible combinations with regard to gender (the first one corresponds to the target voice): i) Female-Female, ii) Male-Male, iii) Female-Male, and iv) Male-Female. Table 4.4 lists the results of performance of our models and the baselines in these setups. Note that all the models perform the best in Female-Male setting and the worst in Male-Female setting (except for Y-Net-m, which performs worst in Male-Male mixtures). This is expected because our training set has larger number of female samples and hence, the models tend to estimate female voice better than the male voice when the female voice is present in the mix. Though the estimation of a voice in a mix with voices from the same gen-

der is difficult, we see that even the performance in the Male-Male setup is better than Male-Female setup because of the tendencies of the models to estimate female voice better. Again, the audio-only model, U-Net, suffers the most from this problem while the audio-visual models can better estimate target male voices thanks to the visual information. While the Y-Net-m is the best model among the models trained with mixtures containing one lead voice, Y-Net-gr remains the best overall model in this work.

Model	Female-Female	Male-Male	Female-Male	Male-Female
U-Net	-1.84	-2.05	0.92	-4.91
LLCP	-0.26	-1.02	0.87	-2.81
Y-Net-m	**3.06**	**1.38**	**5.44**	**1.52**
Y-Net-g	2.96	0.85	5.01	0.23
Y-Net-gr	**7.49**	**3.94**	**7.56**	**6.34**

Table 4.4: Comparing singing voice separation performance based on gender, in the two lead voices setup in the unseen-unheard test sets. The values are in SDR.

Finally, Table 4.5 shows an ablation on the percentage of mixtures with two lead voices in the training set in the case of our best model, Y-Net-gr. The table shows the performance of these models both in the one lead voice and two lead voices setup in the unseen-unheard test set. The increase in the percentage of two lead voice mixtures while training degrades the test results in the one lead voice setup when the volume of the singing voice is reasonable ($\alpha = 1$, as it is the case both in the training and test sets of this ablation). However, as seen in Figure 4.3 (left), when the volume of the target singing voice is low, training the network with two lead voices helps as well – note that Y-Net-gr performs better than Y-Net-g also for one lead voice. When we evaluate the models with mixtures containing two lead voices, we observe a boost in performance of the different Y-Net-gr models (with different percentages of two lead voice mixtures) with respect to that of the Y-Net-g. By considering the

average results of the one lead and two lead voices setups in the unseen-unheard test set (two rightmost columns in Table 4.5), we infer that the best model is the one trained with 50% of the mixtures containing two lead voices. Thus, we consider Y-Net-gr model trained with 50% of mixtures containing two lead voices, as our proposed model as it achieves a good compromise in both scenarios and also in the case of a target voice with a low volume.

Y-Net-gr	One lead voice in test		Two lead voices in test		Average	
Remix %	SDR	SIR	SDR	SIR	SDR	SIR
0 %	**12.47**	**19.71**	2.07	15.49	7.27	17.6
50 %	11.29	19.43	6.41	**17.38**	**8.85**	**18.41**
75 %	11.08	18.78	**6.42**	17.09	8.75	17.93
100 %	9.98	17.73	6.40	16.68	8.19	17.21

Table 4.5: Ablation study on the percentage of mixtures containing two lead voices in the training of the Y-Net-gr model (note that 0% corresponds to the Y-Net-g model). Results on the unseen-unheard test set.

For demos, please visit the project page: `https://ipcv.github.io/Acappella/`.

4.6 Conclusions

This chapter explores the singing voice separation problem from a new perspective, by exploiting both the audio and visual information. We introduce a new dataset of video recordings of *a cappella* solo performances. We also propose a new audio-visual singing voice separation model, based on a U-Net conditioned on the motion of the face landmarks of the target singer. Those landmarks are processed with a spatio-temporal graph convolutional network. Moreover, we present a thorough ablation study of our model, with different variants of the visual network and show how the performance can be boosted in multi-voice cases by adding mixtures with two lead singing voices in the training set. The experiments show how audio-visual methods improve upon audio-only

ones in challenging scenarios when there are multiple overlapping voices or when the target voice has a low volume. The presented model is compared to a state-of-the-art audio-visual speech separation model trained in the new dataset. Our model better exploits the face motion and thus outperforms the baseline models in singing voice separation in the most challenging evaluation setup.

Chapter 5

TRANSFORMER-BASED SPEECH AND SINGING VOICE SEPARATION

5.1 Introduction

Human voice is usually found together with other sounds. Think of people speaking in a cafeteria or in a social gathering, a journalist reporting on the scene, or an artist singing on a stage. In these situations we can find: multiple concurrent speeches, speech with background noise or a single or multiple singing voices with music accompaniment among others. Recalling some cognitive examples from Sec. 1.1, our brain is capable of understanding and concentrating on the voice of interest in a noisy environment [Cherry, 1953]. This cognitive process does not only rely on the hearing. Some works have shown the sight helps to focus on the voice of interest [Golumbic et al., 2013] or to resolve ambiguities in a noisy environment [Ma et al., 2009]. Interestingly, vision perceptually restores auditory spectral dynamics in speech [Plass et al., 2020]. In this chapter, we address the voice separation and enhancement problems from a multimodal perspective, as in the previous chapter, leveraging the motion information extracted from the visual stream to guide the resolution of the problem. To do so, we propose a transformer-based architecture that is competitive against the SOTA network [Gao and Grauman, 2021], that exploit motion and appearance. We study different ways of feeding transformers with audio-visual signals and the similarity between speech and singing voice for AVSS. This chapter is paired with the publication:

"VoViT: Low Latency Graph-based Audio-Visual Voice Separation Transformer" *J.F. Montesinos*, V.S. Kadandale, G. Haro. In *17th European Conference in Computer Vision, ECCV 2022.*

We propose an AV voice separation model that produces state-of-the-art results. It is based on a two-stage approach. The first stage estimates a fairly good separation by combining audio and motion features with a transformer. Motion cues are crucial when the sound mixture contains different predominant voices. As in the previous chapter, we extract those cues with a ST-GCN that processes a sequence of face landmarks. The audio-visual features are aligned in the feature dimension and preserve the time resolution. They are processed by a multimodal spectro-temporal

transformer that estimates the isolated voice corresponding to the target face landmarks. In a second stage, the predominant voice is enhanced by a small audio-only U-Net that takes as input just the pre-estimated audio. The voice of interest is predominant in the first estimation and thus an audio-only network is capable of modelling it and cancelling the sparse and mild interferences present in the pre-estimation. The chapter includes an ablation study of different configurations of the multimodal transformer, its number of blocks and the design of the lead voice enhancer network. The proposed method is compared to state-of-the-art methods in two different scenarios: speech and singing voice separation, showing successful results in both cases. The contributions of this work are several:

i We propose an audio-visual network based on a transformer which performs better than current state-of-the-art models in speech and singing voice separation.

ii We show that a landmark-based approach for extracting motion information can be a lightweight competitive alternative to processing raw video frames.

iii We show how an enhancement stage based on a light network can boost the performance of AV models over larger complex models, reducing the computational cost and the required time for training.

iv We reveal that AV models trained in speech separation do not generalize well enough for the separation of singing voice because of the different voice characteristics in each case and that a dedicated training with singing voice examples clearly boosts the results.

v Our method is an end-to-end gpu-powered system that is capable of isolating a target voice in real-time (including the pre-processing steps).

5.2 Related work

In the last years there has been a fast evolution of deep-learning-based audio-visual works for speech separation and enhancement (we refer the reader to a recent review in [Michelsanti et al., 2021]).

Back in 2016, we can find one of the first works in exploiting visual features for speech enhancement [Wu et al., 2016]. In this work, the authors proposed a CNN to process the visual signal and a fully connected layer to process the raw waveforms. Both modalities were fused by a BiLSTM network. This network had approximately 3M parameters (M for millions), far from the 80M of the most recent work [Gao and Grauman, 2021]. A two-tower stream for processing audio and video features and then fused with a BiLSTM module that predicted complex masks was proposed in [Ephrat et al., 2018].

In [Afouras et al., 2018], a two-step enhancement process was proposed. In the first step, a two-tower stream processed the audio-visual information to extract a binary mask that performed separation on the magnitude spectrogram. Afterwards, the phase of the spectrogram was predicted by passing the estimated magnitude spectrogram together with the noisy phase spectrogam through a 1D-CNN. A similar idea was developed in [Gabbay et al., 2018], where a two-tower stream encoder generated an embedding of audio-visual features from which the enhanced speech spectrogram was recovered. On the other hand, in [Hou et al., 2018] not only the enhanced spectrogram was reconstructed but the input frames as well.

New approaches and explorations different from the two-tower CNNs appeared recently. [Sadeghi and Alameda-Pineda, 2021] joined the scene with variational auto-encoders for speech enhancement. Concurrently, [Wu et al., 2019] developed a time-domain model for speech separation, in contrast to most of the works which usually posed the problem in the time-frequency domain. Multi-channel audio-visual speech separation was addressed in [Gu et al., 2020] in a four-tower stream fashion. The mixture spectrogram was constrained with directional features from the visual stream of the speaker. A temporal CNN extracted visual features

from the lips motion. The audio and visual embeddings were concatenated together with a speaker embedding extracted from the clean audio(s). In [Li and Qian, 2020, Sato et al., 2021, Sun et al., 2020] a different mechanism was used, where the audio-visual fusion was done with an attention module; or in [Xu et al., 2021], where the system was trained in a adversarial manner so that the discriminator modeled the distribution of the clean speech signals. Transformers have been used in audio-only source separation [Zadeh et al., 2019]. Very recently, audio-visual transformers were investigated in [Truong et al., 2021] for main speaker localization and separation of its corresponding audio. In [Tzinis et al., 2021] an audio-visual transformer was used for classification in order to guide an unsupervised source separation model. Finally, in [Chen et al., 2021] a transformer was used for audio-visual synchronisation.

Another interesting proposal is [Chuang et al., 2020], where the authors were concerned about the extra computational cost of processing the visual features and the possible privacy problems arisen from it.

On the other hand, to our knowledge, there are only two works using face landmarks, instead of video frames, for source separation. In [Morrone et al., 2019] they process face landmarks with fully connected layers and then use BiLSTMs to predict the masks for the target source. In [Montesinos et al., 2021], which corresponds to the approach presented in Chapter 4, a U-Net conditioned by a graph convolutional network that processed face landmarks was used for audio-visual singing voice separation.

Most recent algorithms made use of lips motion as well as appearance information, usually implementing cross-modal losses to pull together corresponding audio-visual features. Some interesting works applying that are [Gao and Grauman, 2021, Makishima et al., 2021].

Finally, the work in [Michelsanti et al., 2019] compared different training targets and loss functions for audio-visual speech enhancement.

5.3 Approach

In audio-visual voice separation, given an audio-visual recording with several speaking/singing faces, and other sound sources, the goal is to recover their isolated voices by guiding the voice separation with the visual information present in the video frames. More formally, given the audio signal of each speaker, $x_i(t)$ (where t denotes time), the mixture of sounds can be defined as $x(t) = \sum_i s_i(t) + n(t)$ where $n(t)$ denotes any other sound present in the mixture, i.e. background sounds. Therefore, the task of interest can be defined as the estimation of each individual voice $\hat{s_i}(t)$. In our approach $\hat{x_i}(t) = F(x(t), v_i(t))$, where F is a function represented by a neural network. The network receives the visual information of the speaker of interest, $v_i(t)$, and estimates its isolated voice $\hat{s_i}(t)$.

5.3.1 The AV Voice Separation Network

Our solution comprises of a two-stage neural network that operates in the time-frequency domain. The first stage consists of an AV voice separation network which can isolate the target voice at a good quality. However, this network is the most demanding one in terms of computational cost. To alleviate this, we propose to use downsampled spectrograms in this stage. The second stage consists of a recursive lead voice enhancer network that works with full resolution spectrograms. In Section 5.5.2, we experimentally show that this two-stage design leads to a higher performance than using larger AV models. To achieve this modularity, the networks at both stages are trained independently. The whole model is presented in Fig.5.1.

Stage 1: Audio-Visual Voice Separation. For simplicity, we seek to isolate the voice (denoted by $s(t)$ and its corresponding spectrogram $S[k, l]$) corresponding to a single face at a time. The audio waveform of the mixture, $x(t)$, is transformed into a complex spectrogram $X[k, l]$ applying a STFT. Once the waveform is mapped to the time-frequency domain, we can define a complex mask $M[k, l]$ that allows to recover the spectrogram of the estimated source with a complex product, denoted as

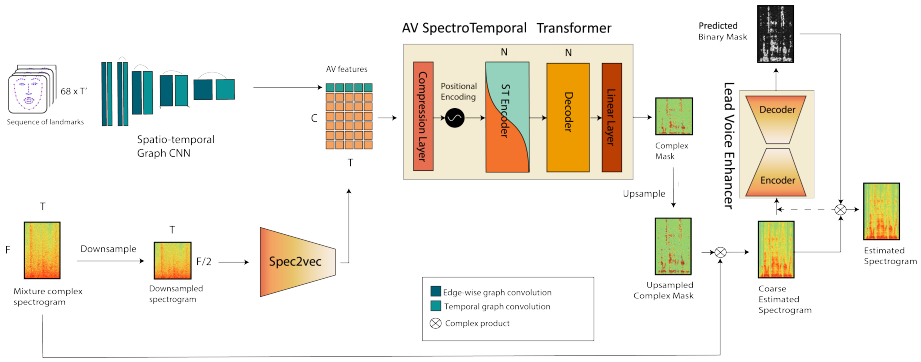

Figure 5.1: Audio-visual voice separation network. Audio and video features are concatenated in the channel dimension before being fed to the transformer.

$\otimes$, that is: $S[k,l] = X[k,l] \otimes M[k,l]$ Then, the goal of the network in the first stage is to estimate the complex mask $\hat{M}[k,l]$. The optimal set of parameters of the network is found by minimising the following loss:

$$\mathcal{L}_1 = \|G \odot (M_b - \hat{M}_b)\|^2$$

where M_b and $\hat{M}_b$ are, respectively, the ground truth and estimated bounded complex masks, $\odot$ denotes the element-wise product, $\|\cdot\|$ is the $L2$-norm and G is a gradient penalty term which weights the time-frequency points of the mask according to the energy of the analogous point in the mixture spectrogram X:

$$G(k,l;X) = \max(\min(\log(1 + \|X[k,l]\|), 10), 10^{-3}). \tag{5.1}$$

Note that, by definition, the ground truth mask M is not bounded. In order to stabilise the training, we bound the complex masks by applying a hyperbolic tangent [Williamson et al., 2015]: $M_b = \tanh M^r + i \tanh M^i$, where M^r and M^i, denote the real and imaginary parts, respectively. The audio waveform of the estimated source can be computed through the inverse STFT of the estimated spectrogram $\hat{S}[k,l] = X[k,l] \otimes \hat{M}[k,l]$.

To solve the AV voice separation problem, we propose to leverage the face motion information present in the video frames of the target person

whose voice we want to isolate. For that, we use a spatio-temporal graph neural network that processes the face landmarks to generate motion features. On the other hand, the audio features are generated by a CNN encoder, denoted as *Spec2vec*. Both audio and motion features preserve the temporal resolution and are concatenated in the channel dimension, then they are fed into a transformer. All the submodules have been carefully designed to achieve a high-performance low-latency neural network.

Spatio-temporal graph CNN: Many AV speech separation or enhancement methods rely on lips motion extracted from raw video frames to guide the task. To reduce the computational and the storage cost of the visual stream, we propose use face landmarks together with a ST-GCN. This network, similar to that in [Montesinos et al., 2021], which corresponds to the previous chapter, however, it was redesigned to preserve the temporal resolution. It consists of a set of blocks which apply a graph convolution over the spatial dimension followed by a temporal convolution. This way we can considerably reduce the amount of data to process and to store, from $96 \times 96 \times 3 \approx 3 \cdot 10^4$ values per frame to $68 \times 3 \approx 10^2$. This supposes a substantial reduction in the storage necessities when working with large audio-visual datasets. For example, *Voxceleb2*'s grayscale ROIs occupy 1Tb, the raw uncompressed dataset occupies several Tb while storing face landmarks only requires 70 Gb.

Spec2vec: It is well known that transformers need proper embeddings to achieve high performance. We use the audio encoder described in the work [Ephrat et al., 2018] to generate embeddings without losing temporal resolution.

AV spectro-temporal transformer: The traditional AV source separation methods comprise of a two-tower stream architecture. We can find two major variants: either encoder-decoder CNNs (usually with a U-Net as backbone) (e.g. [Gao and Grauman, 2021, Gao and Grauman, 2019, Slizovskaia et al., 2021, Zhao et al., 2018, Zhao et al., 2019]) or recurrent neural networks (RNNs), both conditioned on visual features, for example, [Ephrat et al., 2018, Morrone et al., 2019, Wu et al., 2016]. The major drawback of the latter is that RNNs are sequential, introducing bottlenecks in the processing pipeline. As explained in Sec. 2.2.2, trans-

formers appeared as an efficient solution, reaching the same performance than RNNs and CNNs in large datasets. They are trained with a masking system allowing to process all the timesteps of a sequence in parallel. However, these architectures operate sequentially at the time of inference, like the RNNs. To overcome this issue we use an encoder-decoder transformer, which can solve the source separation problem in a single forward pass.

Transformers were originally designed to work with two unimodal signals. We study three different possible configurations for the transformer. The first proposal is to use the transformer as an auto-encoder, being fed with an audio-visual signal directly. This way we ease the task for the transformer as audio and visual features are temporally aligned by construction. Then, it just has to find relationships through the multi-head self-attention. The second proposal is to pass visual features to the encoder and audio features (from the mixture) to the decoder so that the network can find audio-visual interdependencies via multi-head attention. Nevertheless, we hypothesise the dependencies between video and audio are local as audio events mostly occur at the same time than visual events. Lastly, we feed the encoder with an audio-visual signal and the decoder with the ground-truth separated audio. Note that this model is slower than previous ones as the model runs recurrently at inference time, going from a time complexity of $\mathcal{O}(n)$ to $\mathcal{O}(n^2)$ where n is the length of the sequence. From the ablation study in Section 5.5.1 and Table 5.1, we conclude that the best model is the first one, i.e. the one that uses an audio-visual signal as input, we denote it as AV ST-transformer.

We design our AV ST-transformer encoder upon the findings of a work in BSS, [Zadeh et al., 2019]. The AV ST-transformer has 512 model features across 8 heads. We tried 256 features but it works worse. The compression layer is nothing but a fully connected layer followed by a Gaussian Error Linear Unit (GELU) [Hendrycks and Gimpel, 2016] activation which maps the C incoming channels to the 512 channels required by the architecture. It is composed by M encoders and M decoders. The encoder is a set of two traditional encoders in parallel, which processes the signal from a temporal and a spectral point of view [Zadeh et al., 2019].

Stage 2: Lead voice enhancer. Although lips motion is correlated with the voice signal and may help in source separation, it is not always accessible or reliable. For example, the scenarios involving a side view of the speaker or a partial occlusion of the face or an out-of-sync audio-visual pair make it challenging to incorporate the lips motion information in a useful way; all such scenarios may appear in unconstrained video recordings. In the previous chapter, we showed that audio-only models tend to predict the predominant voice in a mixture when there is no prior information about the target speaker. Based on this idea, we hypothesise that, if the first stage of the AV voice separation network outputs a reasonable estimation of the target voice, this voice will be predominant in the estimation. Upon this idea, we use an audio-only network which identifies the predominant voice and enhances the estimation without relying on the motion, just on the pre-estimated audio. To do so, we simply use a small U-Net which takes as input the estimated magnitude spectrogram (at its original resolution) and returns a binary mask. The ground truth binary mask can be obtained from the ground truth spectrogram S and the spectogram to be refined, $\hat{S}$, which is the one estimated in the stage 1:

$$M[k,l] = \begin{cases} 1, & \text{if } \|S[k,l]\| \geq \|\hat{S}[k,l] - S[k,l]\|, \\ 0, & \text{otherwise.} \end{cases} \tag{5.2}$$

Notice that the difference $\hat{S}[k,l] - S[k,l]$ are the remaining sources that need to be removed in the refinement stage.

There are different reasons to use binary masks. On the one hand, we found qualitatively, by inspecting the results, that the secondary speaker is often attenuated but not completely removed. In [Grais et al., 2016], the authors show that binary masks are particularly good at reducing interferences. On the other hand, complex masks appeared as an evolution of binary masks and ratio masks, as a way of estimating, not only the magnitude spectrogram, but the phase too. Note that these masking systems usually reconstruct the estimated waveform with the phase of the mixture as they estimate the magnitude only. In our case, the phase has already been estimated by using complex masks in the previous stage. Lastly, by

using binary masks, we are changing the optimisation problem and easing the task since it is simpler to take a binary decision than orienting and modulating a vector.

Note that this refinement network can run recursively, although we empirically found (see Table 5.2) that applying the refinement network once leads to the best results in terms of SDR and a considerable boost in SIR. Further iterations reduce the interferences (at a lesser extent) but at the cost of introducing more distortion.

Let us denote by $\hat{\hat{M}}$ the binary mask estimated by the lead voice enhancer network. We trained this network to optimise a weighted binary cross entropy loss:

$$\mathcal{L}_2 = \sum_{f=1}^{F} \sum_{t=1}^{T} \frac{G(k,l;\hat{S})}{KL} \Big(M[k,l] \log \|\hat{\hat{M}}[k,l]\| + (1 - M[k,l])(1 - \log \|\hat{\hat{M}}[k,l]\|) \Big)$$

where the weights G are defined in Eq. 5.1.

5.3.2 Low-latency data pre-processing

Many audio-visual works rely on expensive pipelines to pre-process data, which makes the proposed systems unusable in a real-world scenario unless a great amount of time is invested in optimisation. Pursuing the real applicability of our model, we curated an end-to-end gpu-powered system which can pre-process (from raw audio and video) and isolate the target voice of 10s of recordings in less than 100ms using floating-point 32 precision, and in less than 50 ms using floating-point 16 precision.

Face landmarks: The most common approach in speech separation is to align the faces in the different frames via 2D face landmark estimation, as in the previous chapter, together with image warping (e.g. [Gao and Grauman, 2021, Kadandale et al., 2022]). This step removes eventual head motions. In order to achieve real-time audio-visual source separation, we estimate the 3D face landmarks using an optimised version of [Guo et al., 2020]. In this chapter, face landmarks have a dual interpretation, as a 3D point cloud and as an undirect graph. Recalling the nomenclature defined in Sec. 2.3.1, face landmarks can be defined as a set of edges, E, and nodes,V, such that $G = (E, V)$, where the nodes are

denoted as $V = \{v_{it} | i = 1, ..., N, t = 1, ..., T\}$. As a point cloud, we can consider landmarks to live in a projective space $v_{it} = (v_{it}^x, v_{it}^y, v_{it}^z, 1) \in \mathbb{P}^3$. Hence, we can define a projective transformation $\mathcal{Q}$, which is a 4×4 non-singular matrix. A special subtype of projective transformation is the rigid transformation (Eq. 5.3), which is constrained to rotations and translations, where $\mathcal{R}$ denotes a rotation matrix, $\mathcal{T}$ is a translation vector and $\mathbf{0}$ a null vector. This transformation allows to freely move and orientate face landmarks on the space.

$$\mathcal{Q} = \begin{pmatrix} \mathcal{R}_{3\times3} & \mathcal{T}_{3\times1} \\ \mathbf{0}_{1\times3} & 1 \end{pmatrix} \tag{5.3}$$

We apply this rigid transformation to map face landmarks to a frontal standard point of view, as depicted in Fig. 5.2. This process is usually called face alignment in the literature. To do so, we manually define a set of nodes v'_{it}, which are used as a template. The relationship between the estimated landmarks and the template is defined as $v'_{it} = \mathcal{Q}v_{it} + n_{it}$, where n models the residuals, e.g., mismatching between the template and target face due to facial expressions and noise from the estimation. An optimal $\mathcal{Q}$ can be found by minimizing the following expression:

$$\arg\min_{\mathcal{R},\mathcal{T}} ||v'_{it} - \mathcal{Q}v_{it}||$$

The solution to find an optimal rotation was proposed in [Kabsch, 1978], whereas the optimal rigid transformation was solved by a different mathematician in [Arun et al., 1987].

Thanks to the 3D information, we can recover lips motion from side views by estimating 3D landmarks, as shown in Fig. 5.2. Finally, we drop the depth coordinate and consider just the first two spatial coordinates in the nodes of the graph.

Audio: Waveforms are re-sampled to 16384 Hz. Then, we compute a STFT with a window size of 1022 and a hop length of 256. This leads to a $512 \times 64\,n$ complex spectrogram where n is the duration of the waveform in seconds. To reduce the computational cost of both training and inference we downsample the spectrogram in the frequency dimension by 2 in Stage 1.

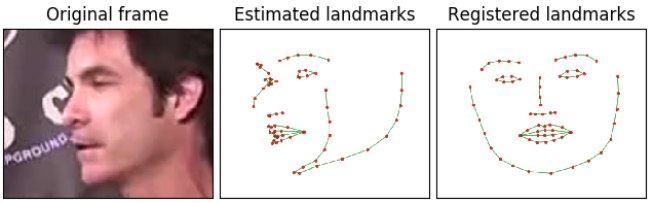

Figure 5.2: Frame example from *Voxceleb2* [Chung et al., 2018] with partial occlusions. Thanks to the landmark estimation together with the registration we can estimate the unoccluded lips.

5.4 Datasets

Experiments are carried out in two different datasets: *Voxceleb2*, a dataset of celebrities speaking in a broad range of scenarios [Chung et al., 2018]; and *Acappella*, a dataset of solo-singing videos presented in the previous chapter and corresponding to [Montesinos et al., 2021]. We also consider *Audioset* [Gemmeke et al., 2017] and *MUSDB18* [Rafii et al., 2017] for sampling extra audio sources that can be added to the singing voice signal as accompaniment.

Voxceleb2 contains 1 million utterances, most of them of a duration between 4 and 6 seconds, consisting of celebrities covering a wide range of ethnicities, professions and ages. The dataset is formed by in-the-wild videos that include several challenging scenarios, such as: different lightning, side-face views, motion blur and poor image quality. They also span across different scenarios like red carpets, stadiums, public speeches, etc. The dataset provides a test set which contains both, seen-heard and unseen-unheard speakers together. From this test set we selected the unseen-unheard samples and curated two different subsets. The first one, denoted as *unheard-unseen wild test set* consists of 1,000 samples randomly selected, reflecting the aforementioned challenges. The second one, denoted as *unheard-unseen clean test set*, is a subset of 1,000 samples, from which 500 of them have a high-quality content with the following characteristics: frontal or almost frontal point of view, low

background noise and perceptual image quality above the average of the dataset. The samples were selected manually from the whole unseen-unheard test set, trying to include as many different speakers as possible. The target voice is sampled from the subset of 500 high-quality videos in the *clean set*, while the second voice is sampled from the rest of 500 videos. This way we ensure that the video content is good enough to estimate motion features from it and that the ground truth separated audio is reliable, in the sense that it does not contain background sounds that may produce unfounded performance metrics.

Acappella is a 46-hours dataset of a cappella solo singing videos. The videos are divided in four language categories: English, Spanish, Hindi and others. These videos are recorded in a frontal view with no occlussions. It also provides two test sets: the seen-heard test set and the unseen-unheard test set. The former contains videos sampled from the same singers and in the same languages than the training set, whereas the latter contains recordings sampled from new singers in the four language categories plus some new languages. In the test set all the categories are equally represented across languages and gender. This way the algorithms can be tested in challenging real-world scenarios.

Audioset [Gemmeke et al., 2017] is an in-the-wild large-scale dataset of audio events across more than 600 categories. We gathered the categories related to the human voice and some typical accompaniments. These categories are: acappella, background music, beatboxing, choir, drum, lullaby, rapping, theremin, whistling and yodelling.

Finally, *MUSDB18* [Rafii et al., 2017] is an audio-only dataset of 150 full-track songs of different styles that includes original sound sources.

5.5 Experiments

The experiments were carried out in a single RTX 3090 GPU. Each experiment takes around 20 days of training. We used SGD with 0.8 momentum, 10^{-5} weight decay and a learning rate of 0.01. The metrics used for comparing results are SDR and SIR [Vincent et al., 2006].

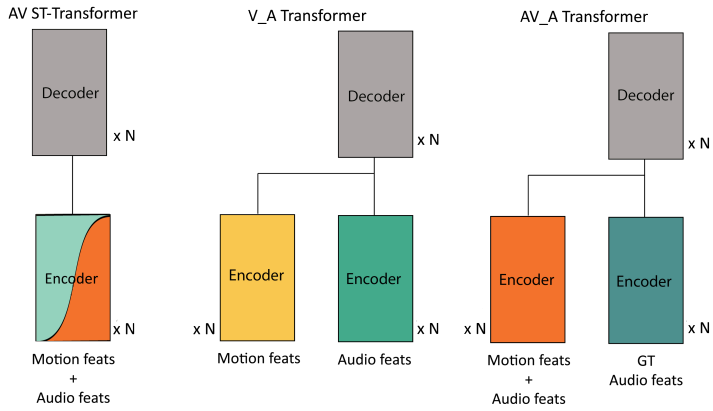

Figure 5.3: Three proposed ways to feed a transformer with an audio-visual signal. Left: audio-visual signal, middle: video to the encoder and audio mixture to the decoder, right: audio-visual signal to the encoder and clean audio to the decoder.

5.5.1 Audio-visual transformer

In this experiment, we compare three different versions of the transformer (shown in Fig. 5.3) in the *Acappella* dataset. The goal is two-fold: i) Compare the proposed architecture against Y-Net, the network proposed in the previous chapter, which was the SOTA model in singing voice separation; and ii) compare the performance of different transformers for the task of singing voice separation.

For the sake of comparison, we train our models the same way as in Chapter 4. In short, we create artificial mixtures of 4s of duration by mixing a voice sample from *Acappella* together with an accompaniment sample sourced either from *Audioset* or *MUSDB18*.

Additionally, a second voice sample from *Acappella* is added 50% of the times. This results in mixtures that contain one or more voices plus musical accompaniment. For this dataset we take 4s audio excerpts and the corresponding 100 video frames from which we extract the face landmarks.

Results are shown in Table 5.1. From the ablation on the three ver-

sions of the transformer, we can conclude that the AV ST-transformer is the best model in terms of both performance and time complexity. Moreover, it can be observed that the three versions of the transformer greatly outperform the results of Y-Net in terms of SDR, while the AV ST-transformer also outperforms in SIR.

Model	Y-Net	AV ST-transformer	V_A transformer	AV_A transformer
SDR ↑	6.41	**10.63 ± 5.86**	8.64 ± 5.89	9.98 ± 5.70
SIR ↑	17.38	**17.67 ± 7.73**	14.70 ± 7.88	16.11 ± 7.42

Table 5.1: Ablation study: performance of different ways of feeding a transformer with an audio-visual signal and comparison to Y-Net model [Montesinos et al., 2021]. Evaluated in *Acappella's* unseen-unheard test set. Y-Net metrics taken from *Acappella*. In this table $N = 4$ (the number of blocks in the transformers) in order to adapt the number of parameters to the size of *Acappella* dataset.

5.5.2 Speech separation

In Section 5.5.1 we found the AV ST-transformer was the best model in terms of time complexity and performance. All the rremaining experiments will be carried out with this model. Now we consider the task of AV speech separation and work with *Voxceleb2* dataset. We use 2s audio excerpts which correspond to 50 video frames from which we extracted their face landmarks. In this case, we mix two voice samples from *Voxceleb2* which are normalised with respect to their absolute maximum, so that a mixture is $x(t) = (s_1(t) + s_2(t))/2$. This normalisation aims to have two voices which are codominant in the mixture and that the waveforms of the mixtures are bounded between -1 and 1. Note that the former characteristic is not always true as *Voxceleb2* samples are sometimes accompanied by other voices or sorts of interference (clapping, music, etc.). As *Voxceleb2* is a large-scale dataset, and for the sake of comparison, we extended the size of the AV ST-transformer up to 10 encoder blocks and 10 decoder blocks so that the number of parameters of the audio subnet-

work is comparable to that of Visual Voice [Gao and Grauman, 2021]. We tested the performance of each model in the *unheard-unseen wild* test set and in the *unheard-unseen clean* test set (both described in Section 5.4). For each test set we randomly made 500 pairs out of the 1,000 samples, ensuring no sample is used more than once.

Lead Voice Enhancer. The first experiment is an ablation designed to address three main questions. i) Compare two different versions of the lead voice enhancer: the audio backbone of Y-Net [Montesinos et al., 2021], which is a 7M-parameter U-Net; and the audio backbone of Visual Voice, yet another U-Net but with 50M parameters because of a different design. ii) Evaluate the effect of recurrent iterations of the lead voice enhancer. And iii) comparing the results of the 10-block 2-stage AV ST-Transformer against a 18-block 1-stage AV ST-Transformer transformer. The details of this subnetwork are explained in Section 5.3.1. We denote our Voice-Visual Transformer as VoViT (the whole network with two stages) and VoViT-s1 the network without the second stage.

The results are shown in Table 5.2. As we can see, the refinement network improves the results substantially for the 10-block AV ST-Transformer. Successive iterations of the refinement module further reduce the interferences, but the best SDR is achieved with just one iteration. For the lead voice enhancer, we tried two possible audio-only U-Nets: the U-Net from the Y-Net model [Montesinos et al., 2021] and the larger U-Net from Visual Voice [Gao and Grauman, 2021]. A much larger U-Net does not outperform the smaller one by a large margin. Interestingly, we can observe that adding this module performs better than using the 18-block AV ST-transformer (with around 2 times more parameters). Moreover, this subnetwork can be trained within a day, whereas the 18-block transformer required around a month to train. The reasons behind the lack of improvement of the 18-block transformer are unknown. We observed a phenomena similar to the so called "double descent" [Nakkiran et al., 2020] while training the 10-block transformer, which may be indicative of a complex optimisation process which is worsened in the 18-block case exceeding our computational resources. In the same line, we trained a larger graph convolutional network, comparable in number of parameters to the

motion subnetwork of Visual Voice, however the performance dropped. From this ablation, we can conclude that a 10-block AV ST-transformer with a small U-Net as lead voice enhancer is the best option in terms of performance-latency trade-off.

		Wild test set	
		SDR ↑	SIR ↑
10-block	VoViT-s1	9.68	15.75
	VoViT (VV in stage 2, $r = 1$)	**10.05**	18.30
	VoViT (VV in stage 2, $r = 2$)	9.77	**19.38**
	VoViT (YN in stage 2, $r = 1$)	10.03	18.18
	VoViT (YN in stage 2, $r = 2$)	9.78	19.09
	18-block VoViT-s1	9.27	15.53

Table 5.2: Ablation of different variants of the refinement stage and number of blocks in the transformer of the first stage. VoViT-s1 stands for the model with just the first stage, r stands for the number of recurrent passes in stage 2. For the stage 2 we considerered both, the Visual Voice's UNet (VV) [Gao and Grauman, 2021] and the Y-Net's UNet (YN) [Montesinos et al., 2021].

Comparison to state-of-the-art methods. Next we are going to compare the 10-block AV ST-Transformer to a state-of-the-art AV speech separation model and audio baselines in the *Voxceleb2* dataset. The Visual Voice network [Gao and Grauman, 2021] is the current state of the art in speech separation. This network uses 2.55s excerpts, the corresponding 64 video frames cropped around the lips and an image of the whole face of the target speaker. Apart from using lips motion features, it extracts cross-modal face-voice embeddings that complement the motion features and are especially useful when the motion is not reliable or when the appearance of the speakers is different. We also compare the results against Y-Net as it is one of the few papers proposing face landmarks. The original work uses 4s excerpts. As around 160k samples for *Voxceleb2* are shorter, we just adapted the model for working with 2s samples.

Numerical results are shown in Table 5.3. The 10-block VoViT out-

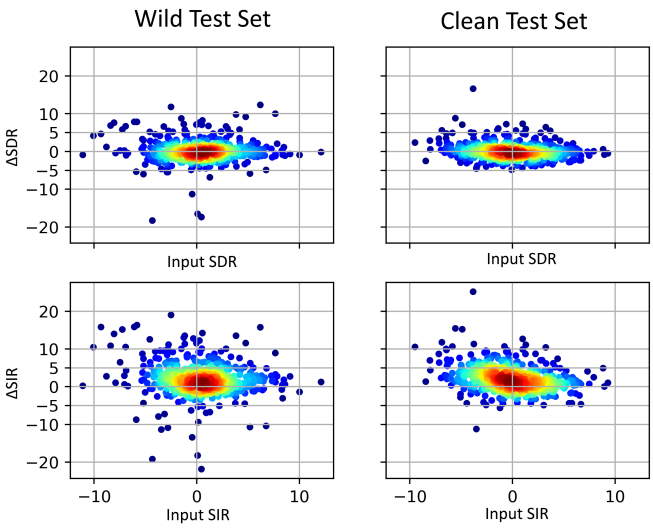

Figure 5.4: Scatter plot showing the difference in SDR and SIR, ΔSDR and ΔSIR, as functions of the SDR and SIR of the input mixture in the unseen-unheard wild and clean test sets. The difference is: $\Delta SDR = SDR(\text{VoViT}) - SDR(\text{Visual Voice})$ so a positive value means VoViT outperforms Visual Voice.

	# parameters		Wild Test set		Clean Test set	
	Visual Net.	Whole Net.	SDR ↑	SIR ↑	SDR ↑	SIR ↑
Visual Voice Audio-only	–	46.14	7.7	13.6	–	–
Face Filter [Chung et al., 2020b]	–	–	2.53	–	–	–
The conversation [Afouras et al., 2018]	–	–	8.89	14.8	–	–
Visual Voice Motion-only	9.14	55.28	9.94	17	–	–
Y-Net [Montesinos et al., 2021]	1.42	9.7	5.29 ± 5.06	8.45 ± 6.8	5.86 ± 4.78	9.25 ± 6.44
Visual Voice [Gao and Grauman, 2021]	20.38	77.75	9.92 ± 3.56	16.11 ± 4.8	10.18 ± 3.36	16.49 ± 4.5
VoViT	1.42	58.2	$\mathbf{10.03 \pm 3.35}$	$\mathbf{18.18 \pm 4.72}$	$\mathbf{10.25 \pm 2.61}$	$\mathbf{18.65 \pm 3.8}$

Table 5.3: Evaluation on *Voxceleb2* unheard-unseen test sets (mean $\pm$ standard deviation). VoViT stands for our model with the 10-block AV ST-Transformer with the Y-Net's UNet backbone as the lead voice enhancer. Number of parameters in millions. Results in the first block are taken from the original papers.

performs all the previous AV speech separation models. Compared to Visual Voice, it achieves a much better SIR and slightly better SDR, both for the wild and clean test sets. In particular, for the clean test set, when the motion cues are more reliable, our model has a much lower standard deviation. Some aspects need to be taken into account:

- The face landmark extractor has been trained with higher quality videos than the ones in *Voxceleb2*. On the contrary, the Visual Voice video network has been trained specifically for *Voxceleb2*.

- Our visual subnetwork, the graph CNN, has 10 times less parameters than its counterpart in Visual Voice.

- Apart from motion cues, Visual Voice takes also into account speaker appearance features which are correlated with voice features, and which can be crucial in poor quality videos where lip motion is unreliable.

Fig. 5.4 shows SDR and SIR differences between VoViT and Visual Voice in two different test sets: the *wild* and the *clean* set. Each plot is a scatter plot where each point corresponds to a 2s long mixture. As it can be observed, our method especially outperforms Visual Voice in SIR while in SDR both methods have a comparable performance. In order to assess the significance of the results of Table 5.3, we calculated the p-values with respect to the Visual Voice results. Only the improvement on SIR is significant ($p < 0.05$). While the improvement from stage 1

to 2 (Table 5.2) is significant both in SDR and SIR. In the *wild test set* there are a few samples where our model performs worse than Visual Voice. Those correspond to samples where the audio and video are extremely unsynchronised or samples where the lip motion is mispredicted and the network separates the other speaker. In those cases, the Visual Voice model might be able to alleviate the situation either by relying on the appearance features to guide the separation or by using the motion information present in the raw video despite its poor quality (e.g. blur, compression artefacts, lack of sharpness). There are no such cases in the *clean set*, as those type of samples were filtered out.

Speed of inference In Table 5.4, the required time to carry out a forward pass is shown. The inference can be carried out fully end-to-end in a GPU, including face landmark estimation. VoViT performs way beyond real time when running on GPUs in PyTorch 1.10 which makes it suitable for cloud computing or local services. It is remarkable there is still a huge margin to improve. Future PyTorch releases shall include fused kernels and optimized routines for transformers, as well as native complex32 support, which will speed up the current system.

	Preprocessing	Inference		Preprocessing + Inference
		Graph Network	Whole model	
VoViT-s1	17.95	4.50	52.21	82.18
VoViT	17.95	4.55	57.45	93.31
VoViT-s1 fp16	10.94	2.88	30.47	52.43
VoViT fp16	10.94	2.86	34.18	46.14

Table 5.4: Latency estimation for the different variants of VoViT. Average of 10 runs, batch size 100. Device: Nvidia RTX 3090. GPU utilization $>$98%, memory on demand. Two forward passed done to warm up. Timing corresponds to ms to process 10s of audio.

5.5.3 Singing voice separation

In this last experiment we consider the task of singing voice. We are interested in exploring how transferable models trained for speech separation are to the case of singing voice. Since speech models were trained with two voices and no extra sounds and in *Voxceleb2*, which contains mainly English, we restricted to similar types of mixtures in singing voice. In particular, we create mixtures of two singers in English from the unseen-unheard test set of *Acappella*, with no accompaniment. Table 5.5 compares the results of models trained directly with samples of singing voice (top block of results in Table 5.5) versus models trained with speech samples (bottom block). In the case of singing voice we used our model with just the first stage and a 4-block AV ST-transformer. We observe that dedicated models for singing voice perform largely better than models trained for speech. This may be explained to particular differences between a speaking and a singing voice. For example, vowels are much more sustained in singing voice, there is much less coarticulation of consonants with surrounding vowels and vibrato is not present in speech. Moreover, singing voice contains varying pitches covering a wider frequency range.

Model	SDR ↑	SIR ↑
Y-Net [Montesinos et al., 2021]	11.08 ± 7.51	17.18 ± 9.68
VoViT-s1 (4 blocks)	**14.85** ± **7.87**	**21.06** ± **9.69**
VoViT-s1	3.89 ± 9.28	5.89 ± 11.15
VoViT	4.04 ± 10.30	**7.21** ± **13.26**
Visual Voice [Gao and Grauman, 2021]	**4.52** ± **8.64**	7.03 ± 7.11

Table 5.5: Singing voice separation. Mixtures of two singers with no additional accompaniment from the test set unseen-unheard (only samples in English) of *Acappella*. Results in top block: models trained directly with samples of singing voice; bottom block: models trained with speech samples.

5.6 Conclusions and Future work

In this work we present a lightweight audio-visual source separation method which can process 10s of recordings in less than 0.1s in an end-to-end GPU powered manner. Besides, the method shows competitive results to the state-of-the-art in reducing distortions while clearly outperforming in reducing interferences. We show that face landmarks are computationally cheaper alternatives to raw video and help to deal with large-scale datasets. For the first time, we evaluate AV speech separation systems in singing voice, showing empirically that the characteristics of the singing voice differ substantially from the ones of speech.

As future work we would like to explore lighter and faster embedding generators for the transformer and different optimisations in its architecture which leads to a fast and powerful system.

Part II

Audio-visual Inpainting

Chapter 6

AUDIO-VISUAL SPEECH INPAINTING

6.1 Introduction

Speech is one of the most common multimodal events in our daily life. Thanks to the expansion of the internet, we are exposed to a lot of speech signals from digital content as well: news, social networks, virtual meetings and video calls. Sometimes, the audio stream is corrupted due to, e.g., muted microphones, external noises, transmission losses or transient signals. One option is to estimate the lost audio information, saving content creators the time to re-make their videos or avoiding a speaker to repeat a sentence. The process of restoring the corrupted audio signal is known as *audio inpainting* [Adler et al., 2012]. Carrying out such a restoration for long segments of corrupted audio (>200ms) is not a simple task, as there is no prior information about the missing content. There are several ways to address the problem. From an audio-only (AO) perspective, the work in [Ebner and Eltelt, 2020] relies on a generative adversarial network approach to generate realistic speech content for a gap size up to 500 ms. In [Chang et al., 2019], an encoder-decoder architecture is used to inpaint the audio in both time-frequency and time domain for segments up to 250 ms. The works in [Marafioti et al., 2019] and [Kegler et al., 2020] propose a similar idea operating only in the time-frequency domain for gaps up to 64 ms and 400 ms, respectively.

There are works using additional modalities as cues to guide the inpainting process. This allows to inpaint larger gaps. For example, works like [Borsos et al., 2022] uses text to guide the inpainting process of audio gaps up to 1000 ms, relying on transformers and contrastive learning. In [Morrone et al., 2021], video information is extracted from face landmarks to inpaint gaps up to 1600 ms. Similarly, we present a deep learning model which can restore long gaps of speech, leveraging the visual information of the speaker, a task known as audio-visual speech inpainting (AVSI). This chapter is paired with the publication:

"Speech Inpainting: Context-based Speech Synthesis Guided by Video"
J.F. Montesinos, D. Michelsanti, G. Haro, J. Jensen, *Under review 2022.*

The contribution of this paper is two-fold:

i We propose a transformer architecture that analyzes a time-frequency representation of the corrupted audio signal and the corresponding uncorrupted visual information to synthesize intelligible speech even for a long corrupted audio segment

ii We show that speech inpainting can benefit from using high-level visual features extracted with the AV-HuBERT [Shi et al., 2022], whose effectiveness for related tasks has previously been reported.

6.2 Approach

6.2.1 Signal Model

Let $x[t]$ be a discrete-time acoustic speech signal and $X[k, l]$ be the corresponding STFT, where k and l indicate a frequency and a time indices, respectively, as described in Sec. 2.1.2. Furthermore, let $\mathcal{A} \in \mathbb{R}^{K \times L}$ denote a magnitude spectrogram matrix defined from the element-wise absolute values of the elements in X. Then, the inpainted signal, $\mathcal{Q} \in \mathbb{R}^{K \times L}$, can be defined as $\mathcal{Q} = \mathcal{M} \odot \mathcal{A} + (\mathbf{1} - \mathcal{M}) \odot \hat{\mathcal{A}}$, where $\odot$ indicates the element-wise product, $\hat{A} \in \mathbb{R}^{K \times L}$ denotes an estimated speech STFT magnitude matrix and $\mathcal{M} \in \mathbb{R}^{K \times L}$ is a binary mask that provides the position of the corrupted region of the spectrogram [Morrone et al., 2021, Paulino and Hounie, 2020]). For the binary mask matrix, we assume that the i-th column consists of ones if the i-th column of A is uncorrupted and zeros otherwise.

6.2.2 Proposed Framework

AVSI leverages the video stream to improve speech inpainting, by providing information about the acoustic speech content within the corrupted region. Our processing pipeline is divided into four different stages: feature extraction, multi-modal fusion, inpainting process and waveform reconstruction. The whole process is depicted in Fig. 6.1.

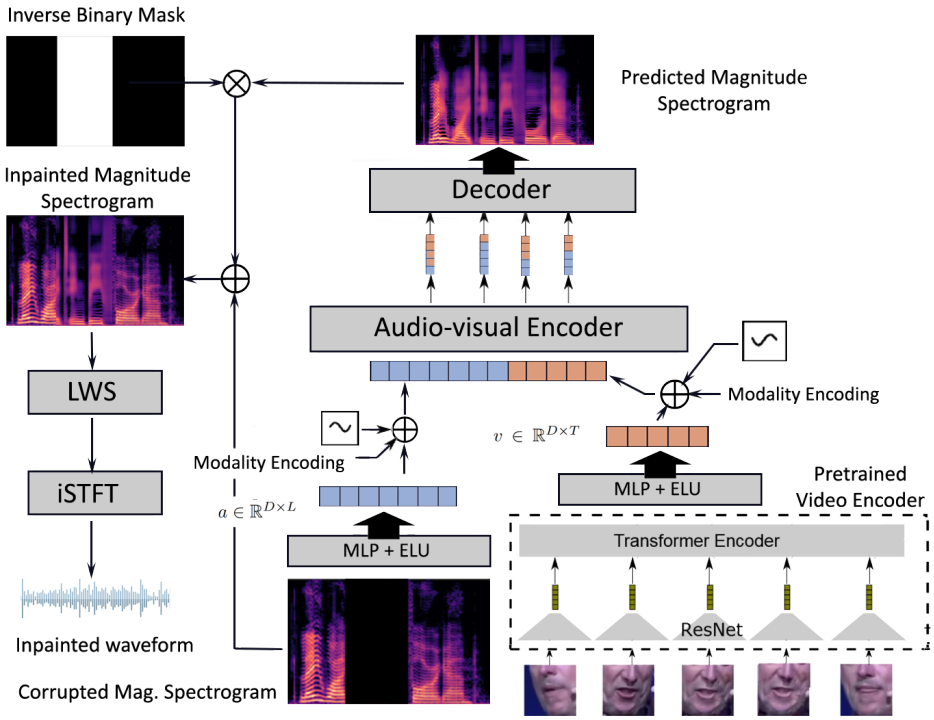

Figure 6.1: Proposed audio-visual model. The pre-trained video encoder corresponds to [Shi et al., 2022].

In the feature extraction stage, we extract high-level visual features using the AV-HuBERT's [Shi et al., 2022] video encoder, which processes the sequence of video frames using a ResNet [He et al., 2016] followed by a transformer encoder to model the temporal dependencies. In addition, we use a simple multi-layer perceptron (MLP) with exponential linear unit (ELU) activation on top, leading to a signal $v \in \mathbb{R}^{D \times T}$, where D is the dimensionality of our embeddings and T the amount of frames. In order to extract learned acoustic features, we use a similar MLP that takes as input the masked spectrogram $X \odot \mathcal{M}$, resulting in a signal $a \in \mathbb{R}^{D \times L}$.

In the multi-modal fusion stage, the goal is to fuse the acoustic and visual features, learning the relationship between both. To do so, we rely

on a six-block transformer encoder that ingests an audio-visual (AV) embedding. We construct the AV embedding by concatenating both modalities temporally. Since the transformer is unaware of the position or the modality type of each element in the sequence, we sum a positional encoding (pe) that reflects the temporal sorting of the elements in the sequence, as in Chapter 5, and a modality encoding (me) that transmits whether each element is an acoustic or a visual feature [Chen et al., 2021], obtaining:

$$\begin{aligned} a \oplus v = pe_a + me_a + a \oplus \\ pe_v + me_v + v, \end{aligned} \tag{6.1}$$

where $\oplus$ denotes the concatenation of two sequences. The AV signal lives in the space $\mathbb{R}^{D \times (T+L)}$. Alternatively, channel-wise-stacked AV embeddings can be used. Nevertheless, we empirically found that, in case of an out-of-sync AV stream, the former concatenation results in predictions which are shifted in time, whereas in the latter case, the system collapses and generates mumbling. An out-of-sync AV stream may occur due to software or hardware issues: codecs, latency, missing frames and it is frequent in low-quality videos.

In the inpainting stage, we use a seven-block transformer that processes the high-level features generated by the encoder to predict the signal $\hat{\mathcal{A}}$. At this stage, the transformer's role is two-fold: It has to act as an auto-encoder, i.e. reconstruct the uncorrupted segment of the audio, and it has to inpaint the corrupted segment.

In the waveform reconstruction stage, we estimate the phase of the predicted spectrogram using Local Weighted Sums (LWS), an algorithm proposed at [Le Roux et al., 2010] and then compute the inverse STFT to recover the waveform, as done in [Morrone et al., 2021].

6.3 Experiments

6.3.1 Audio-Only and Audio-Visual Baselines

We compare the proposed AVSI model against the previous state-of-the-art AV model, proposed in [Morrone et al., 2021], and against the AO version of our model. In the AV baseline [Morrone et al., 2021], the authors propose a framewok whose core is a stack of three Bi-LSTM layers fed with an AV signal. As acoustic features, they use normalized log magnitude spectrograms, while the visual features are landmark-based motion vectors. In order to fuse the acoustic and the visual features via concatenation, they upsample the visual features to the sampling rate of the spectrogram. Then, they compute the first temporal derivative of the landmarks to obtain motion vectors and they concatenate them to the spectrogram. They minimize the mean squared error of the predicted log magnitude spectogram with respect to the ground-truth one in the corrupted segment. Note that this is different from our setup, as we apply the loss on the whole predicted signal, not only in the corrupted segment.

To explore the benefits of using the additional modality of the video stream, we also train our model in an AO setup, i.e., without visual information as input.

6.3.2 The Dataset

We train our model and the baselines using the *Grid Corpus*, proposed in [Cooke et al., 2006], which is an AV dataset consisting of 33 speakers recorded in a controlled environment with a chroma screen as background, a frontal point of view, a controlled lightning and a small vocabulary. Each video is 3 s long, recorded at 25 fps for the video and at 50 kHz for the audio.

We split the dataset into training, validation, and testing, as done in [Morrone et al., 2021]. We corrupt the data with gaps of a duration between 160 and 1600 ms. During training, the corrupted segments are distributed randomly along each sample in a batch. During validation we

apply the same logic so that the distribution of the validation set is as close as possible to the one of training. During testing, we run the system in 5 different setups: a random distribution of the gaps, as described before; corrupted segments with a gap of size 160 ms, 400 ms, 800 ms and 1600 ms. The GRID sentences typically include initial and trailing silence regions. When corrupting the speech signals, we ensure that the entire corrupted segment is located in the speech active parts of the GRID sentences.

6.3.3 Loss, Data Pre-Processing and Model Setup

We downsample the waveforms to 16 kHz. We compute the STFT with a hop size of 256, and a Hanning window of length 512. To process the video, we crop the mouth region, resizing the resulting frames to 96×96. Lastly, we extract the visual features as described in Section 6.2.2.

The transformer ingests 512-element embeddings across 8 heads. The dimensionality of the transformer's feed-forward layer is 1024. We use GELU [Hendrycks and Gimpel, 2016] activation for the transformer and Exponential Linear Unit (ELU) [Clevert et al., 2016] everywhere else. We train the model with a batch size of 10, a learning rate of $1e^{-4}$ and the ADAM optimizer. As loss function we use a weighted Mean Absolute Error (MAE):

$$\alpha \cdot MAE(\hat{\mathcal{A}}^c, \mathcal{A}^c) + \beta \cdot MAE(\hat{\mathcal{A}}^u, \mathcal{A}^u),\ \alpha, \beta \geq 0$$

where the superindices *c* and *u* denote corrupted and uncorrupted parts, respectively. We set $\alpha > \beta$, so that the network is forced to focus on the inpainting task, as it is much harder than the auto-encoding task (we use $\alpha = 10$ and $\beta = 1$).

6.4 Results

6.4.1 Performance Measures

We evaluate our model using three metrics: the MAE between the magnitude spectrogram and the ground-truth within the corrupted speech region; *STOI* [Taal et al., 2011], a speech intelligibility estimate; and *PESQ* [Rix et al., 2001], a speech quality estimate. STOI and PESQ scores lie between -1 and 1, and -0.5 and 4.5, respectively. While lower MAE scores corresponds to a lower reconstruction loss, for PESQ and STOI, the higher the better.

Since it is not possible to use STOI and PESQ for signals shorter than a few hundreds ms, we cannot use them only on the corrupted part. Therefore, we compute the scores for the whole signal. This lowers the sensitivity of the metrics, especially when inpainting short segments.

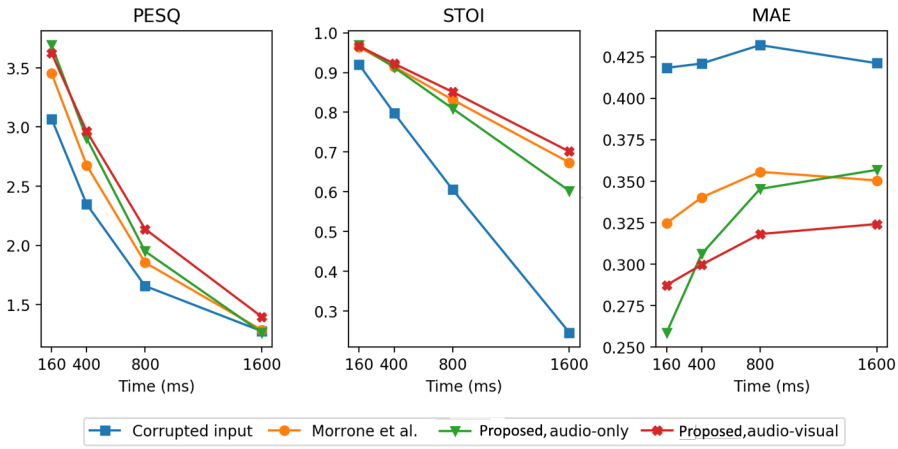

Figure 6.2: Comparison of performance vs corruption duration evaluated in the test set (see Sec. 6.3.2).

6.4.2 General Performance

As our goal is to develop a system capable of dealing with corrupted segments of any duration, rather than training a system specifically for each gap length, in Table 6.1, we report the overall performance in the test set for a distribution of segment durations that matches that of the training stage. As it can be clearly seen, the proposed AV model is not only better than its AO counterpart, but it also outperforms the previous state-of-the-art AV model [Morrone et al., 2021].

	PESQ ↑	STOI ↑	MAE ↓
Corrupted input	1.78	0.58	0.43
[Morrone et al., 2021]	1.98	0.79	0.39
Proposed, audio-only	2.07	0.79	0.34
Proposed, audio-visual	**2.21**	**0.84**	**0.31**

Table 6.1: Performance scores averaged across test set. Corrupted segment lenghts sampled from a uniform distribution.

6.4.3 Performance vs Segment Duration

From Table 6.1, we can notice that the performance of the AV baseline, [Morrone et al., 2021], is worse than the proposed AO model. As AO methods are good at inpainting short gaps, we carried out an analysis of the performance of each model against the corrupted segment duration. The results are shown in Fig. 6.2. Considering the MAE values, we can see that they don't change significantly for segments larger than 800 ms. We hypothesize that the uncorrupted audio is used to determine the voice characteristics and the speech continuity in the boundary of the corrupted segment, while the rest is purely generated from the visuals. Besides, the relative MAE between the reconstructed segments of 1600 ms and 160 ms (27% for the AO model and around 10% for the AV models) shows the effectiveness of the AV methods, as the MAE degradation of the AO model is much higher.

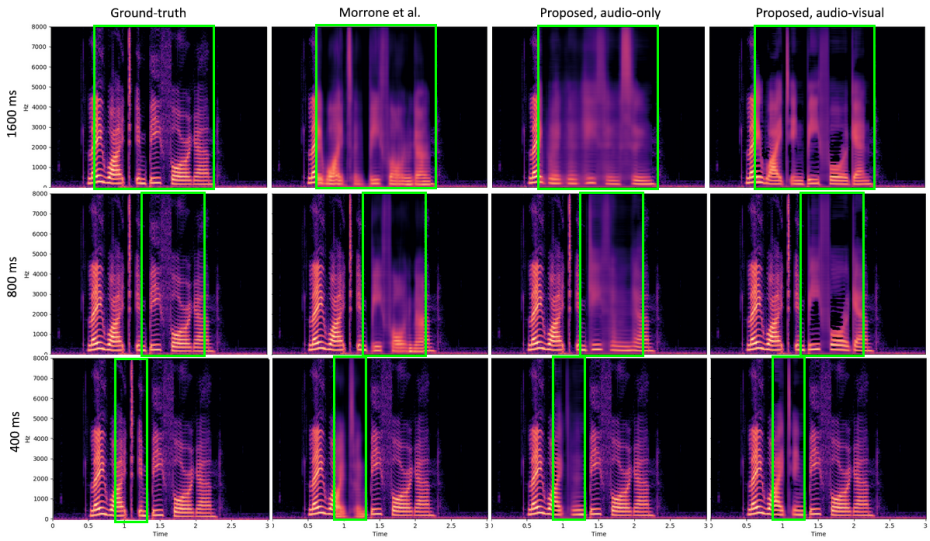

Figure 6.3: Sentence *lwib4a* for speaker 34 in the test set. Transcription: *"lay white in b four again"*. The region within the green square indicates the corrupted area. In practice that region is set to zero as input to the network.

Analysing the results for each segment duration, the performances of the proposed AV and AO models are roughly similar when considering corrupted segments of 160 ms.

On contrary, for corrupted segments of 400 ms, the proposed AV model is better in intelligibility and perceived quality. Nevertheless, the AO model is still very effective. In Fig. 6.3 we can observe how the spectrogram predicted by the AO model is similar to that of the AV model, even the harmonics are better-defined than in the AV baseline's spectrogram.

For corrupted segments of 800 ms and 1600 ms, the proposed AV model is the best. At this point, the AO model is no longer capable of estimating the content of the sentence. It just generates a kind of mumbling, either as a consequence of inpainting the sample with certain energy bands that matches the harmonics of the voice or as an at-

tempt to mimic sentences learned from the dataset. If we consider PESQ, we can see that, for segments of 1600 ms, the scores for the models tend to collapse to a single point. Our hypothesis is that, for such a long gap, the speech context is almost non-existent (see Fig. 6.3), therefore the task becomes close to speech reconstruction from silent videos [Michelsanti et al., 2021], for which speech characteristics of unknown speakers, that are important for PESQ, cannot be easily estimated using only the video information.

6.5 Conclusions and Future Work

This paper presented a new state-of-the-art AVSI model that can inpaint long gaps, up to 1600 ms, for unseen-unheard speakers. We tested our model in the GRID [Cooke et al., 2006] dataset and showed that it outperforms its audio-only counterpart for gaps larger than 160 ms, and the previous state-of-the-art approach. In addition, we showed that the visual features extracted from the AV-HuBERT network encode enough information to guide the inpainting process. One of the limitations of the proposed and the existing AVSI approaches is that the mapping between phonemes and visemes is not bijective, namely, a single viseme may correspond to many phonemes [Fisher, 1968]. For example, the sentences *"elephant juice"* and *"I love you."* share the same visemes. To overcome this limitation, we can incorporate additional information to the approach, such us context information about the scenario or language models.

Chapter 7

CONCLUSIONS

7.1 Overview

The goal of this thesis is to develop deep learning models for audio-visual speech and music source separation, as well as for speech inpainting, overcoming different challenges that affect the field: lack of data, storage cost, computational cost or privacy. In summary, we proposed state-of-the-art models for AV voice separation and AV speech inpainting. We released two different datasets for AV music source separation and AV singing voice separation. We proved the suitability of deep-learning embeddings and face landmarks as a light and efficient representation of videos containing human faces, where motion-based approaches have proven to be powerful enough and more identity preserving. We also compared audio-visual approaches against audio-only ones, concluding AV approaches are superior in noisy or complex acoustic scenarios for the task of source separation, and superior for long-gap speech inpainting. The thesis was structured in the following way:

- In Chapter 1 we motivated the importance of the audio-visual field. We talked about the inspiration from cognitive studies, where multiple benefits from multimodal perception were shown. On the way, we pointed out the importance of developing algorithms with

perception mechanisms similar to humans, which is relevant for human-machine interaction. We also highlighted the necessity of self-supervised algorithms to process the millions of hours of unprocessed recordings, and reviewed some practical applications such as conferencing systems, speaker speech enhancement and diarization. Lastly, we exposed different relevant challenges in the audio-visual field.

- In Chapter 2 we provided the technical foundations required to understand different concepts and tools used in this thesis: acoustic and visual representations, relevant deep neural networks, foundations on graphs and sound source separation.

- In Chapter 3 we curated a new dataset for audio-visual music information retrieval tasks, *Solos* [Montesinos et al., 2020]. This dataset consists of a collection of YouTube IDs of different solo musical performances with a set of chamber instruments matching that of University of Rochester Multi-Modal Music Performance Dataset [Li et al., 2019], together with relevant time-stamps and body skeletons. We evaluated its usefulness for training sound source separation models with different AV and audio-only baselines.

- In Chapter 4 we explored the singing voice separation problem and the suitability of face landmarks as replacement of raw video by proposing a new deep learning model, showing face landmarks are superior to raw video in small datasets. We created *Acappella*, a new dataset of a cappella solo singing videos. We dug into the relationship between AV voice separation and language, showing the performance of the proposed model evaluated in unseen languages is similar to the model's performance evaluated on languages existing in the dataset. Lastly, we analyzed the role of the visual features in two difference scenarios: in presence of a single leading voice with accompaniment and in presence of two leading voices with accompaniment. In the former case, we concluded visual features play an important role when the accompaniment is predominant. In

the latter case, we concluded visually-guided models perform better than audio-only models.

- In Chapter 5 we tackled the speech separation problem, proving that models using face landmarks are competitive against state-of-the-art AV speech separation models. We proposed a new two-stage model that is trained by optimizing two different losses and masks: complex mask and binary masks, which boost the results at a cheap cost. We also compared different transformer-based architectures to process AV signals and evaluated the performance of speech models to separate singing voice signals, showing that there is a domain gap arisen from the differences between speech and a singing voice.

- In Chapter 6 we faced the AV speech inpainting problem. We used visual features extracted from AV HuBERT [Shi et al., 2022] to inpaint long audio gaps, up to 1.6s. We compared our proposed AV model against its audio-only counterpart, which highlighted the importance of visuals, as the performance of audio-only models decreases with the gap duration.

7.2 Limitations and future work

The most relevant limitations of AV methods are the mapping between audio and video is not always correlated, and the presence of ambiguities among both modalities. That is the case, for example, in speech, where the correspondence between phonemes and visemes is not bijective and a single viseme can correspond to several phonemes, as noticed in Sec. 6.5. This would require additional information to disentangle the mapping, such as context information or language models.

As future work, it would be worth studying how to improve model robustness against out-of-sync recordings, which is a frequent issue due to different leakages in the processing and transmission pipelines. Besides, it would be interesting to explore hybrid approaches for source separation. Hybrid approaches are those that process a time representation as well as

a time-frequency representation, making profit from both representations. Hybrid approaches were briefly overviewed in Sec. 2.4.3. In addition, diffusion models could help to accomplish a complete re-synthesis of the isolated sources as diffusion models run iteratively generating low frequency information and high frequency information, achieving great details. While we explored singing voice separation, we did not dig into voices singing in unison, namely, voices singing the same lyrics at the same time. Here, motion cues are not enough to carry out the separation, and additional information is required.

Lastly, in case of inpainting, it would be relevant to extend the studies to speech inpainting when both modalities are corrupted, creating models that can inpaint both modalities. In this scenario, we can identify three possible plots: non-overlapped corruption, partially-overlapped corruption and fully overlapped corruption. In case of partially-overlapped corruption, inpainting both modalities should increase cross-modal coherence and robustness. Due to dataset constrains, the inpainted gaps were as large as 1.6s. Thus, creating datasets with larger excerpts is necessary to explore very long gap inpainting. Lastly, it would be interesting to carry out human evaluation tests and compare different phase reconstruction methods and vocoders, which can achieve higher naturalness in the inpainted voice.

7.3 List of contributions

In-proceedings publications

All the publications are accompanied by code, video presentations, weights, and explanatory project pages for reproducibility.

"Solos: A Dataset for Audio-Visual Music Source Separation and Localization" J.F. Montesinos, O. Slizovskaia, G. Haro. In *In 22st IEEE International Workshop on Multimedia Signal Processing, MMSP 2020.*
Project Page: `www.juanmontesinos.com/Solos/`
Source Code: `github.com/JuanFMontesinos/Solos`

"Multi-channel U-Net for Music Source Separation" J.F. Montesinos, V.S. Kadandale, G. Haro. In *In 22st IEEE International Workshop on Multimedia Signal Processing, MMSP 2020.*
Project Page: `vskadandale.github.io/multi-channel-unet/`
Source Code: `github.com/vskadandale/multichannel-unet-bss`

"A cappella: Audio-visual Singing Voice Separation" J.F. Montesinos, V.S. Kadandale, G. Haro. In *32nd British Machine Vision Conference, BMVC 2021.*
Project Page: `ipcv.github.io/Acappella/`
Source Code: `github.com/JuanFMontesinos/Acappella-YNet`

"VoViT: Low Latency Graph-based Audio-Visual Voice Separation Transformer" J.F. Montesinos, V.S. Kadandale, G. Haro. In *17th European Conference in Computer Vision, ECCV 2022.*
Project Page: `ipcv.github.io/VoViT/`
Source Code: `github.com/JuanFMontesinos/VoViT`

"VocaLiST: An Audio-Visual Synchronisation Model for Lips and Voices" V.S. Kadandale, J.F. Montesinos, G. Haro. In *Interspeech 2022.*
Project Page: `ipcv.github.io/VocaLiST/`
Source Code: `github.com/vskadandale/vocalist`

"Speech Inpainting: Context-based Speech Synthesis Guided by Video" J.F. Montesinos, D. Michelsanti, G. Haro. Zheng-Hua Tao, J. Jensen. *Under review* 2022.
Project Page: `ipcv.github.io/avsi/`

Workshop contributions

"Estimating Individual A Cappella Voices in Music Videos with Singing Faces" V.S. Kadandale, J.F. Montesinos, G. Haro In *Sight and Sound*

workshop, IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021.

"Audio-visual Voice Separation Transformer" J.F. Montesinos, V.S. Kadandale, G. Haro. In *Sight and Sound workshop, IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2022.*

"Synchronisation of lips and voices" J.F. Montesinos, V.S. Kadandale, G. Haro. In *Sight and Sound workshop, IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2022.*

Datasets

Solos: A dataset of instrumentalists playing solo excerpts for a wide range of instruments [Montesinos et al., 2020].
Project Page: `https://www.juanmontesinos.com/Solos/`

Acappella: A dataset of a cappella solo singing videos [Montesinos et al., 2021].
Project Page: `https://ipcv.github.io/Acappella/`

It’s been a pleasure to write this thesis. Thanks for reading.

Juan

Bibliography

[Adler et al., 2012] Adler, A., Emiya, V., Jafari, M. G., Elad, M., Gribonval, R., and Plumbley, M. D. (2012). Audio inpainting. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(3):922–932.

[Afouras et al., 2018] Afouras, T., Chung, J. S., and Zisserman, A. (2018). The conversation: Deep audio-visual speech enhancement. In *Interspeech*, pages 3244–3248.

[Aiello and Dunbar, 1993] Aiello, L. C. and Dunbar, R. I. (1993). Neocortex size, group size, and the evolution of language. *Current anthropology*, 34(2):184–193.

[Appleton et al., 1975] Appleton, J. H., Perera, R., and Luening, O. (1975). *The development and practice of electronic music*. Prentice Hall.

[Arandjelović and Zisserman, 2018] Arandjelović, R. and Zisserman, A. (2018). Objects that sound. In *Proceedings of the IEEE European Conference on Computer Vision*.

[Arun et al., 1987] Arun, K. S., Huang, T. S., and Blostein, S. D. (1987). Least-squares fitting of two 3-d point sets. *IEEE Transactions on pattern analysis and machine intelligence*, (5):698–700.

[Bank et al., 2020] Bank, D., Koenigstein, N., and Giryes, R. (2020). Autoencoders. *CoRR*, abs/2003.05991.

[Benesty et al., 2008] Benesty, J., Sondhi, M. M., Huang, Y., et al. (2008). *Springer handbook of speech processing*, volume 1. Springer.

[Borsos et al., 2022] Borsos, Z., Sharifi, M., and Tagliasacchi, M. (2022). Speechpainter: Text-conditioned speech inpainting. *Interspeech*.

[Cao et al., 2019] Cao, Z., Hidalgo Martinez, G., Simon, T., Wei, S., and Sheikh, Y. A. (2019). Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

[Chang et al., 2019] Chang, Y.-L., Lee, K.-Y., Wu, P.-Y., Lee, H.-y., and Hsu, W. (2019). Deep long audio inpainting. *arXiv preprint arXiv:1911.06476*.

[Charbonneau et al., 2013] Charbonneau, G., Véronneau, M., Boudrias-Fournier, C., Lepore, F., and Collignon, O. (2013). The ventriloquist in periphery: Impact of eccentricity-related reliability on audio-visual localization. *Journal of Vision*, 13(12):20–20.

[Chen et al., 2021] Chen, H., Xie, W., Afouras, T., Nagrani, A., Vedaldi, A., and Zisserman, A. (2021). Audio-visual synchronisation in the wild. *32nd British Machine Vision Conference (BMVC)*.

[Chen, 2001] Chen, T. (2001). Audiovisual speech processing. *IEEE signal processing magazine*, 18(1):9–21.

[Cherry, 1953] Cherry, E. C. (1953). Some experiments on the recognition of speech, with one and with two ears. *The Journal of the acoustical society of America*, 25(5):975–979.

[Chuang et al., 2020] Chuang, S.-Y., Tsao, Y., Lo, C.-C., and Wang, H.-M. (2020). Lite audio-visual speech enhancement. In *Interspeech*.

[Chung et al., 2018] Chung, J. S., Nagrani, A., and Zisserman, A. (2018). Voxceleb2: Deep speaker recognition. In *Interspeech*.

[Chung et al., 2020a] Chung, S.-W., Choe, S., Chung, J. S., and Kang, H.-G. (2020a). FaceFilter: Audio-Visual Speech Separation Using Still Images. In *Interspeech*, pages 3481–3485.

[Chung et al., 2020b] Chung, S.-W., Choe, S., Chung, J. S., and Kang, H.-G. (2020b). Facefilter: Audio-visual speech separation using still images. *arXiv preprint arXiv:2005.07074*.

[Cinelli et al., 2021] Cinelli, L. P., Marins, M. A., Da Silva, E. A. B., and Netto, S. L. (2021). *Variational Methods for Machine Learning with Applications to Deep Networks*. Springer.

[Clevert et al., 2016] Clevert, D., Unterthiner, T., and Hochreiter, S. (2016). Fast and accurate deep network learning by exponential linear units (elus). In *4th International Conference on Learning Representations, ICLR*.

[Cole et al., 2017] Cole, F., Belanger, D., Krishnan, D., Sarna, A., Mosseri, I., and Freeman, W. T. (2017). Synthesizing normalized faces from facial identity features. In *Proceedings of the IEEE Con. on Computer Vision and Pattern Recognition*, pages 3703–3712.

[Cooke et al., 2006] Cooke, M., Barker, J., Cunningham, S., and Shao, X. (2006). An audio-visual corpus for speech perception and automatic speech recognition. *The Journal of the Acoustical Society of America*, 120(5):2421–2424.

[Devlin et al., 2018] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

[Diniz et al., 2010] Diniz, P. S., Da Silva, E. A., and Netto, S. L. (2010). *Digital signal processing: system analysis and design*. Cambridge University Press.

[Doire and Okubadejo, 2019] Doire, C. S. J. and Okubadejo, O. (2019). Interleaved multitask learning for audio source separation with independent databases. *ArXiv*, abs/1908.05182.

[Dronkers and Ogar, 2004] Dronkers, N. and Ogar, J. (2004). Brain areas involved in speech production. *Brain*, 127(7):1461–1462.

[Dumoulin et al., 2018] Dumoulin, V., Perez, E., Schucher, N., Strub, F., Vries, H. d., Courville, A., and Bengio, Y. (2018). Feature-wise transformations. *Distill*, 3(7):e11.

[Ebner and Eltelt, 2020] Ebner, P. P. and Eltelt, A. (2020). Audio inpainting with generative adversarial network. *arXiv preprint arXiv:2003.07704*.

[Ellis, 1996] Ellis, D. P. W. (1996). *Prediction-driven computational auditory scene analysis*. PhD thesis, Massachusetts Institute of Technology.

[Ephrat et al., 2018] Ephrat, A., Mosseri, I., Lang, O., Dekel, T., Wilson, K., Hassidim, A., Freeman, W. T., and Rubinstein, M. (2018). Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation. volume 37, pages 1–11. ACM New York, NY, USA.

[Fernandez-Lopez et al., 2017] Fernandez-Lopez, A., Martinez, O., and Sukno, F. M. (2017). Towards estimating the upper bound of visual-speech recognition: The visual lip-reading feasibility database. In *IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pages 208–215.

[Fisher, 1968] Fisher, C. G. (1968). Confusions among visually perceived consonants. *Journal of speech and hearing research*, 11(4):796–804.

[Gabbay et al., 2018] Gabbay, A., Shamir, A., and Peleg, S. (2018). Visual speech enhancement. In *Interspeech*, pages 1170–1174. ISCA.

[Gan et al., 2020] Gan, C., Huang, D., Zhao, H., Tenenbaum, J. B., and Torralba, A. (2020). Music gesture for visual sound separation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10478–10487.

[Gao and Grauman, 2019] Gao, R. and Grauman, K. (2019). Co-separating sounds of visual objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3879–3888.

[Gao and Grauman, 2021] Gao, R. and Grauman, K. (2021). Visualvoice: Audio-visual speech separation with cross-modal consistency. In *2021 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Gemmeke et al., 2017] Gemmeke, J. F., Ellis, D. P., Freedman, D., Jansen, A., Lawrence, W., Moore, R. C., Plakal, M., and Ritter, M. (2017). Audio set: An ontology and human-labeled dataset for audio events. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 776–780.

[Golumbic et al., 2013] Golumbic, E. Z., Cogan, G. B., Schroeder, C. E., and Poeppel, D. (2013). Visual input enhances selective speech envelope tracking in auditory cortex at a "cocktail party". *The Journal of Neuroscience*, 33:1417 – 1426.

[Grais et al., 2016] Grais, E. M., Roma, G., Simpson, A. J., and Plumbley, M. (2016). Combining mask estimates for single channel audio source separation using deep neural networks. In *Interspeech*.

[Griffin and Lim, 1984] Griffin, D. and Lim, J. (1984). Signal estimation from modified short-time fourier transform. *IEEE Transactions on acoustics, speech, and signal processing*, 32(2):236–243.

[Gu et al., 2020] Gu, R., Zhang, S.-X., Xu, Y., Chen, L., Zou, Y., and Yu, D. (2020). Multi-modal multi-channel target speech separation. *IEEE Journal of Selected Topics in Signal Processing*, 14(3):530–541.

[Guo et al., 2020] Guo, J., Zhu, X., Yang, Y., Yang, F., Lei, Z., and Li, S. Z. (2020). Towards fast, accurate and stable 3d dense face alignment. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK*, pages 152–168.

[Haim et al., 2022] Haim, N., Vardi, G., Yehudai, G., Shamir, O., and Irani, M. (2022). Reconstructing training data from trained neural networks. *arXiv preprint arXiv:2206.07758*.

[He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

[Hendrycks and Gimpel, 2016] Hendrycks, D. and Gimpel, K. (2016). Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

[Hershey and Movellan, 2000] Hershey, J. R. and Movellan, J. R. (2000). Audio vision: Using audio-visual synchrony to locate sounds. In *Advances in neural information processing systems*, pages 813–819.

[Hofman et al., 1998] Hofman, P., Riswick, J., and Opstal, J. (1998). Relearning sound localization with new ears. nature neuroscience 1:417-421. *Nature neuroscience*, 1:417–21.

[Hou et al., 2018] Hou, J.-C., Wang, S.-S., Lai, Y.-H., Tsao, Y., Chang, H.-W., and Wang, H.-M. (2018). Audio-visual speech enhancement using multimodal deep convolutional neural networks. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2(2):117–128.

[Hsu et al., 2021] Hsu, W.-N., Bolte, B., Tsai, Y.-H. H., Lakhotia, K., Salakhutdinov, R., and Mohamed, A. (2021). Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460.

[Hyvärinen and Oja, 2000] Hyvärinen, A. and Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5):411–430.

[IBM, 1991] IBM, M. (1991). Multimedia programming interface and data specifications 1.0.

[Isola et al., 2017] Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976.

[Jansson et al., 2017] Jansson, A., Humphrey, E., Montecchio, N., Bittner, R., Kumar, A., and Weyde, T. (2017). Singing voice separation with deep U-Net convolutional networks. In *International Society for Music Information Retrieval Conference (ISMIR)*, pages 23–27.

[Jespersen, 2013] Jespersen, O. (2013). *Language: Its nature, development, and origin*. Routledge.

[Kabsch, 1978] Kabsch, W. (1978). A discussion of the solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A*, 34(5):827–828.

[Kadandale et al., 2022] Kadandale, V. S., Montesinos, J. F., and Haro, G. (2022). Vocalist: An audio-visual synchronisation model for lips and voices. In *Proceedings of Interspeech*, pages 3128–3132.

[Kadandale et al., 2020] Kadandale, V. S., Montesinos, J. F., Haro, G., and Gómez, E. (2020). Multi-channel u-net for music source separation. In *IEEE International Workshop on Multimedia Signal Processing (MMSP)*.

[Kadve, 2016] Kadve, A. R. (2016). Trade of between ssd and hdd. *International Journal for Research in Applied Science and Engineering Technology*, 4:473–475.

[Kawaler and Czyżewski, 2019] Kawaler, M. and Czyżewski, A. (2019). Database of speech and facial expressions recorded with optimized face motion capture settings. *Journal of Intelligent Information Systems*, 53(2):381–404.

[Kay et al., 2017] Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P.,

et al. (2017). The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*.

[Kegler et al., 2020] Kegler, M., Beckmann, P., and Cernak, M. (2020). Deep speech inpainting of time-frequency masks. *Interspeech*.

[Kidron et al., 2005] Kidron, E., Schechner, Y. Y., and Elad, M. (2005). Pixels that sound. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 88–95.

[Kim et al., 2018] Kim, C., Shin, H. V., Oh, T.-H., Kaspar, A., Elgharib, M., and Matusik, W. (2018). On learning associations of faces and voices. In *Asian Conference on Computer Vision*, pages 276–292. Springer.

[Kingma and Ba, 2014] Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

[Kiper, 2016] Kiper, D. (2016). The physics of sound. `https://homepages.wmich.edu/~hillenbr/206/ac.pdf`.

[Korbar et al., 2018] Korbar, B., Tran, D., and Torresani, L. (2018). Cooperative learning of audio and video models from self-supervised synchronization. In *Advances in Neural Information Processing Systems*, volume 31.

[Le Roux et al., 2010] Le Roux, J., Kameoka, H., Ono, N., and Sagayama, S. (2010). Fast signal reconstruction from magnitude stft spectrogram based on spectrogram consistency. In *Proceedings DAFx*, volume 10, pages 397–403.

[Lee et al., 2021] Lee, J., Chung, S.-W., Kim, S., Kang, H.-G., and Sohn, K. (2021). Looking into your speech: Learning cross-modal affinity for audio-visual speech separation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1336–1345.

[Leman, 2017] Leman, M. (2017). The Interactive Dialectics of Musical Meaning Formation. In Lesaffre, M., Maes, P.-J., and Leman, M., editors, *The Routledge Companion to Embodied Music Interaction*, pages 13–21. Routledge.

[Leong et al., 2020] Leong, M., Prasad, D., Lee, Y. T., and Lin, F. (2020). Semi-cnn architecture for effective spatio-temporal learning in action recognition. *Applied Sciences*, 10:557.

[Li et al., 2017a] Li, B., Dinesh, K., Duan, Z., and Sharma, G. (2017a). See and listen: Score-informed association of sound tracks to players in chamber music performance videos. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2906–2910. IEEE.

[Li et al., 2019] Li, B., Dinesh, K., Xu, C., Sharma, G., and Duan, Z. (2019). Online audio-visual source association for chamber music performances. *Transactions of the International Society for Music Information Retrieval*, 2(1).

[Li et al., 2019] Li, B., Liu, X., Dinesh, K., Duan, Z., and Sharma, G. (2019). Creating a multitrack classical music performance dataset for multimodal music analysis: Challenges, insights, and applications. *IEEE Transactions on Multimedia*, 21(2):522–535.

[Li et al., 2021a] Li, B., Wang, Y., and Duan, Z. (2021a). Audiovisual singing voice separation. *Transactions of the International Society for Music Information Retrieval*, 4(1).

[Li et al., 2017b] Li, B., Xu, C., and Duan, Z. (2017b). Audiovisual source association for string ensembles through multi-modal vibrato analysis. *Proceedings of Sound and Music Computing (SMC)*.

[Li and Qian, 2020] Li, C. and Qian, Y. (2020). Deep audio-visual speech separation with attention mechanism. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 7314–7318.

[Li et al., 2021b] Li, T., Chen, J., Hou, H., and Li, M. (2021b). Sams-net: A sliced attention-based neural network for music source separation. In *International Symposium on Chinese Spoken Language Processing (ISCSLP)*, pages 1–5.

[Li et al., 2022] Li, T., Liu, Y., Owens, A., and Zhao, H. (2022). Learning visual styles from audio-visual associations. *arXiv preprint arXiv:2205.05072*.

[Liu et al., 2018] Liu, G., Si, J., Hu, Y., and Li, S. (2018). Photographic image synthesis with improved u-net. In *2018 Tenth International Conference on Advanced Computational Intelligence (ICACI)*, pages 402–407.

[Lu et al., 2018] Lu, W., Chen, Z., Li, L., Cao, X., Wei, J., Xiong, N., Li, J., and Dang, J. (2018). Watermarking based on compressive sensing for digital speech detection and recovery †. *Sensors*, 18.

[Ma et al., 2009] Ma, W. J., Zhou, X., Ross, L. A., Foxe, J. J., and Parra, L. C. (2009). Lip-reading aids word recognition most in moderate noise: a bayesian explanation using high-dimensional feature space. *PloS one*, 4(3):e4638.

[Makishima et al., 2021] Makishima, N., Ihori, M., Takashima, A., Tanaka, T., Orihashi, S., and Masumura, R. (2021). Audio-visual speech separation using cross-modal correspondence loss. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6673–6677. IEEE.

[Mao et al., 2016] Mao, X., Shen, C., and Yang, Y.-B. (2016). Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In *Advances in neural information processing systems*, pages 2802–2810.

[Marafioti et al., 2019] Marafioti, A., Perraudin, N., Holighaus, N., and Majdak, P. (2019). A context encoder for audio inpainting.

IEEE/ACM Transactions on Audio, Speech, and Language Processing, 27(12):2362–2372.

[Mercea et al., 2022] Mercea, O.-B., Riesch, L., Koepke, A., and Akata, Z. (2022). Audio-visual generalised zero-shot learning with cross-modal attention and language. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10553–10563.

[Meseguer-Brocal and Peeters, 2019] Meseguer-Brocal, G. and Peeters, G. (2019). Conditioned-u-net: Introducing a control mechanism in the u-net for multiple source separations. In *International Society for Music Information Retrieval Conference (ISMIR)*.

[Meseguer-Brocal and Peeters, 2020] Meseguer-Brocal, G. and Peeters, G. (2020). Content based singing voice source separation via strong conditioning using aligned phonemes. In *International Society for Music Information Retrieval Conference (ISMIR)*.

[Michelsanti et al., 2019] Michelsanti, D., Tan, Z.-H., Sigurdsson, S., and Jensen, J. (2019). On training targets and objective functions for deep-learning-based audio-visual speech enhancement. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8077–8081. IEEE.

[Michelsanti et al., 2020] Michelsanti, D., Tan, Z.-H., Zhang, S.-X., Xu, Y., Yu, M., Yu, D., and Jensen, J. (2020). An overview of deep-learning-based audio-visual speech enhancement and separation. *arXiv preprint arXiv:2008.09586*.

[Michelsanti et al., 2021] Michelsanti, D., Tan, Z.-H., Zhang, S.-X., Xu, Y., Yu, M., Yu, D., and Jensen, J. (2021). An overview of deep-learning-based audio-visual speech enhancement and separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:1368–1396.

[Mira et al., 2022] Mira, R.-S. C., Haliassos, A., Petridis, S., Schuller, B. W., and Pantic, M. (2022). Svts: Scalable video-to-speech synthesi. *Interspeech*, pages 1836–1840.

[Montesinos et al., 2021] Montesinos, J. F., Kadandale, V. S., and Haro, G. (2021). A cappella: Audio-visual singing voice separation. In *32nd British Machine Vision Conference, BMVC*.

[Montesinos et al., 2022a] Montesinos, J. F., Kadandale, V. S., and Haro, G. (2022a). Vovit: Low latency graph-based audio-visual voice sseparation transformer. In *European Conference on Computer Vision (ECCV)*.

[Montesinos et al., 2022b] Montesinos, J. F., Michelsanti, D., Haro, G., Tan, Z.-H., and Jensen, J. (2022b). Speech inpainting: Context-based speech synthesis guided by video. In *Under review*.

[Montesinos et al., 2020] Montesinos, J. F., Slizovskaia, O., and Haro, G. (2020). Solos: A dataset for audio-visual music analysis. In *22st IEEE International Workshop on Multimedia Signal Processing, MMSP 2020, Tampere, Finland, September 21-24, 2020*. IEEE.

[Morrone et al., 2019] Morrone, G., Bergamaschi, S., Pasa, L., Fadiga, L., Tikhanoff, V., and Badino, L. (2019). Face landmark-based speaker-independent audio-visual speech enhancement in multi-talker environments. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6900–6904. IEEE.

[Morrone et al., 2021] Morrone, G., Michelsanti, D., Tan, Z.-H., and Jensen, J. (2021). Audio-visual speech inpainting with deep learning. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6653–6657.

[Nakkiran et al., 2020] Nakkiran, P., Kaplun, G., Bansal, Y., Yang, T., Barak, B., and Sutskever, I. (2020). Deep double descent: where bigger models and more data hurt. *Journal of Statistical Mechanics: Theory and Experiment*, 2021.

[Ngoc et al., 2020] Ngoc, Q. T., Lee, S., and Song, B. C. (2020). Facial landmark-based emotion recognition via directed graph neural network. *Electronics*, 9(5):764.

[Nguyen et al., 2020] Nguyen, V.-N., Sadeghi, M., Ricci, E., and Alameda-Pineda, X. (2020). Deep variational generative models for audio-visual speech separation. *arXiv preprint arXiv:2008.07191*.

[Oh et al., 2019] Oh, T.-H., Dekel, T., Kim, C., Mosseri, I., Freeman, W. T., Rubinstein, M., and Matusik, W. (2019). Speech2face: Learning the face behind a voice. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 7539–7548.

[O'shaughnessy, 1987] O'shaughnessy, D. (1987). *Speech communications: Human and machine*. Universities press.

[Owens and Efros, 2018] Owens, A. and Efros, A. A. (2018). Audio-visual scene analysis with self-supervised multisensory features. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

[Papadopoulos et al., 2021] Papadopoulos, K., Kacem, A., Shabayek, A., and Aouada, D. (2021). Face-gcn: A graph convolutional network for 3d dynamic face identification/recognition. *arXiv preprint arXiv:2104.09145*.

[Parekh et al., 2017] Parekh, S., Essid, S., Ozerov, A., Duong, N. Q., Pérez, P., and Richard, G. (2017). Guiding audio source separation by video object information. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 61–65.

[Paulino and Hounie, 2020] Paulino, I. R. and Hounie, I. (2020). Paco and paco-dct: Patch consensus and its application to inpainting. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5775–5779.

[Perez et al., 2018] Perez, E., Strub, F., De Vries, H., Dumoulin, V., and Courville, A. (2018). Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

[Petermann et al., 2020] Petermann, D., Chandna, P., Cuesta, H., Bonada, J., and Gomez, E. (2020). Deep learning based source separation applied to choir ensembles. *arXiv preprint arXiv:2008.07645*.

[Plass et al., 2020] Plass, J., Brang, D., Suzuki, S., and Grabowecky, M. (2020). Vision perceptually restores auditory spectral dynamics in speech. *Proceedings of the National Academy of Sciences*, 117(29):16920–16927.

[Platz and Kopiez, 2012] Platz, F. and Kopiez, R. (2012). When the eye listens: A meta-analysis of how audio-visual presentation enhances the appreciation of music performance. *Music Perception*, 30(1):71–83.

[Rafii et al., 2017] Rafii, Z., Liutkus, A., Stöter, F.-R., Mimilakis, S. I., and Bittner, R. (2017). The MUSDB18 corpus for music separation.

[Rahimi et al., 2022] Rahimi, A., Afouras, T., and Zisserman, A. (2022). Reading to listen at the cocktail party: Multi-modal speech separation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10493–10502.

[Rix et al., 2001] Rix, A. W., Beerends, J. G., Hollier, M. P., and Hekstra, A. P. (2001). Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 2, pages 749–752.

[Ronneberger et al., 2015] Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.

[Rouard et al., 2022] Rouard, S., Massa, F., and Défossez, A. (2022). Hybrid transformers for music source separation. *arXiv preprint arXiv:2211.08553*.

[Sadeghi and Alameda-Pineda, 2021] Sadeghi, M. and Alameda-Pineda, X. (2021). Mixture of inference networks for vae-based audio-visual speech enhancement. *IEEE Transactions on Signal Processing*, 69:1899–1909.

[Samuel et al., 2020] Samuel, D., Ganeshan, A., and Naradowsky, J. (2020). Meta-learning extractors for music source separation. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 816–820.

[Sato et al., 2021] Sato, H., Ochiai, T., Kinoshita, K., Delcroix, M., Nakatani, T., and Araki, S. (2021). Multimodal attention fusion for target speaker extraction. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pages 778–784. IEEE.

[Schulkin and Raglan, 2014] Schulkin, J. and Raglan, G. B. (2014). The evolution of music and human social capability. *Frontiers in neuroscience*, 8:292.

[Shams and Seitz, 2008] Shams, L. and Seitz, A. R. (2008). Benefits of multisensory learning. *Trends in Cognitive Sciences*, 12(11):411–417.

[Shi et al., 2022] Shi, B., Hsu, W.-N., Lakhotia, K., and Mohamed, A. (2022). Learning audio-visual speech representation by masked multimodal cluster prediction. *International Conference on Learning Representations (ICLR)*.

[Shlizerman et al., 2017] Shlizerman, E., Dery, L. M., Schoen, H., and Kemelmacher-Shlizerman, I. (2017). Audio to body dynamics. *2017 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*.

[Slizovskaia et al., 2021] Slizovskaia, O., Haro, G., and Gómez, E. (2021). Conditioned source separation for musical instrument performances. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:2083–2095.

[Slizovskaia et al., 2019] Slizovskaia, O., Kim, L., Haro, G., and Gomez, E. (2019). End-to-end sound source separation conditioned on instrument labels. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 306–310.

[Smaragdis et al., 2006] Smaragdis, P., Raj, B., and Shashanka, M. (2006). A probabilistic latent variable model for acoustic modeling. *Advances in models for acoustic processing, NIPS*, 148:8–1.

[Song et al., 2017] Song, C., Ristenpart, T., and Shmatikov, V. (2017). Machine learning models that remember too much. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 587–601.

[Stoller et al., 2018] Stoller, D., Ewert, S., and Dixon, S. (2018). Wave-u-net: A multi-scale neural network for end-to-end audio source separation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 2391–2395.

[Sun et al., 2020] Sun, Z., Wang, Y., and Cao, L. (2020). An attention based speaker-independent audio-visual deep learning model for speech enhancement. In *International Conference on Multimedia Modeling*, pages 722–728. Springer.

[Taal et al., 2011] Taal, C. H., Hendriks, R. C., Heusdens, R., and Jensen, J. (2011). An algorithm for intelligibility prediction of time–frequency weighted noisy speech. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2125–2136.

[Takahashi et al., 2018] Takahashi, N., Goswami, N., and Mitsufuji, Y. (2018). MMDenseLSTM: An efficient combination of convolutional

and recurrent neural networks for audio source separation. In *International Workshop on Acoustic Signal Enhancement (IWAENC)*, pages 106–110.

[Thompson et al., 2005] Thompson, W. F., Graham, P., and Russo, F. A. (2005). Seeing music performance: Visual influences on perception and experience. *Semiotica*, 2005(156):203–227.

[Tran et al., 2018] Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., and Paluri, M. (2018). A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6450–6459.

[Truong et al., 2021] Truong, T.-D., Duong, C. N., Pham, H. A., Raj, B., Le, N., Luu, K., et al. (2021). The right to talk: An audio-visual transformer approach. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1105–1114.

[Tzinis et al., 2021] Tzinis, E., Wisdom, S., Remez, T., and Hershey, J. R. (2021). Improving on-screen sound separation for open-domain videos with audio-visual self-attention. *arXiv preprint arXiv:2106.09669*.

[Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

[Vincent et al., 2006] Vincent, E., Gribonval, R., and Fevotte, C. (2006). Performance measurement in blind audio source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(4):1462–1469.

[Virtanen, 2007] Virtanen, T. (2007). Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria. *IEEE transactions on audio, speech, and language processing*, 15(3):1066–1074.

[Wang and Kurz, 2022] Wang, Q. and Kurz, D. (2022). Reconstructing training data from diverse ml models by ensemble inversion. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2909–2917.

[Wang et al., 2020] Wang, W., Tran, D., and Feiszli, M. (2020). What makes training multi-modal classification networks hard? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12695–12705.

[Williamson et al., 2015] Williamson, D. S., Wang, Y., and Wang, D. (2015). Complex ratio masking for monaural speech separation. *IEEE/ACM transactions on audio, speech, and language processing*, 24(3):483–492.

[Wu et al., 2019] Wu, J., Xu, Y., Zhang, S.-X., Chen, L.-W., Yu, M., Xie, L., and Yu, D. (2019). Time domain audio visual speech separation. In *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 667–673.

[Wu et al., 2016] Wu, Z., Sivadas, S., Tan, Y. K., Bin, M., and Goh, R. S. M. (2016). Multi-modal hybrid deep neural network for speech enhancement. *arXiv preprint arXiv:1606.04750*.

[Xu et al., 2019] Xu, X., Dai, B., and Lin, D. (2019). Recursive visual sound separation using minus-plus net. In *Proceedings of the International Conference on Computer Vision*, pages 882–891.

[Xu et al., 2021] Xu, X., Wang, Y., Xu, D., Peng, Y., Zhang, C., Jia, J., and Chen, B. (2021). Vsegan: Visual speech enhancement generative adversarial network. *arXiv preprint arXiv:2102.02599*.

[Yan et al., 2018] Yan, S., Xiong, Y., and Lin, D. (2018). Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

[Yu et al., 2017] Yu, F., Koltun, V., and Funkhouser, T. (2017). Dilated residual networks. In *Computer Vision and Pattern Recognition (CVPR)*.

[Zadeh et al., 2019] Zadeh, A., Ma, T., Poria, S., and Morency, L.-P. (2019). Wildmix dataset and spectro-temporal transformer model for monoaural audio source separation. *arXiv preprint arXiv:1911.09783*.

[Zhao et al., 2019] Zhao, H., Gan, C., Ma, W.-C., and Torralba, A. (2019). The sound of motions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1735–1744.

[Zhao et al., 2018] Zhao, H., Gan, C., Rouditchenko, A., Vondrick, C., McDermott, J., and Torralba, A. (2018). The sound of pixels. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 570–586.

[Zhou et al., 2019] Zhou, H., Liu, Z., Xu, X., Luo, P., and Wang, X. (2019). Vision-infused deep audio inpainting. In *The IEEE International Conference on Computer Vision (ICCV)*.

[Zhu and Rahtu, 2020] Zhu, L. and Rahtu, E. (2020). Visually guided sound source separation using cascaded opponent filter network. In *Proceedings of the Asian Conference on Computer Vision*.

[Zhu and Rahtu, 2021] Zhu, L. and Rahtu, E. (2021). Visually guided sound source separation and localization using self-supervised motion representations.

[Zibulevsky and Pearlmutter, 2001] Zibulevsky, M. and Pearlmutter, B. A. (2001). Blind source separation by sparse decomposition in a signal dictionary. *Neural computation*, 13(4):863–882.